SSP 50305 Volume I REV B
January 2002

POIC to Generic User Interface Definition Document, Volume 1

International Space Station Program

Revision B

JANUARY 2002

R S
Russian
Space
Agency

Canadian Space
Agency

Agence spatiale
canadienne

asi
agenzia spaziale
italiana

esa
european space agency

NASDA
宇宙開発事業団

NASA

National Aeronautics and Space Administration
International Space Station Program
Johnson Space Center
Houston, Texas

INTERNATIONAL SPACE STATION PROGRAM

PREFACE

POIC TO GENERIC USER INTERFACE DEFINITION DOCUMENT, VOLUME 1

JANUARY 2002

The contents of this document are intended to be consistent with the tasks and products to be prepared by the International Space Station Program participants as defined in the Concept of Operations and Utilization (COU). This document is under the control of the Ground Segment Control Board (GSCB), with the concurrence of the affected members of the GSCB. Any changes or revisions will be approved by the GSCB.

GROUND SEGMENT CONTROL BOARD NOTICE

INTERNATIONAL SPACE STATION PROGRAM

POIC TO GENERIC USER INTERFACE DEFINITION DOCUMENT, VOLUME 1

REVISION B

JANUARY 2002

/s/ Ernest E. Smith 05/16/2002

Ernest E. Smith DV!4 Date
GSCB Chairman
Ground Systems Integration Office
Mission Operations Directorate
NASA – Johnson Space Center

/s/ Darrell G. Bailey 03/18/2002
Darrell G. Bailey Date
GSCB MSFC Representative
Flight Projects Directorate
NASA – Marshall Space Flight Center

/s/ Rochell Brown 04/08/2002
Rochelle Brown Date
GSCB JSC Representative
NASA – Johnson Space Center

/s/ Roland Luettgens 03/18/2002
Roland Luettgens Date
GSCB ESA Representative

(*See Concurrence Page)*
Claudio Canu Date
GSCB ASI Representative

/s/ Junjiry Nakahara 11/26/2001
Junjiro Nakahara Date
GSCB NASDA Representative

/s/ Real Palardy	03/19/2002
Real Palardy GSCB CSA Representative	Date
/s/ Boris Motsulyev	02/08/2002
Boris Motsulyev GSCB RSA Representative	Date
/s/ Ted Sobchak	03/21/2002
Ted Sobchak GSCB GSFC Representative	Date

Data Quality Assurance:

/s/ Paul Iademarco	03/19/2002
Paul Iademarco OL Configuration Management International Space Station Program Office NASA – JSC	Date

INTERNATIONAL SPACE STATION PROGRAM

POIC TO GENERIC USER INTERFACE DEFINITION DOCUMENT, VOLUME 1

REVISION B

JANUARY 2002

CONCURRENCE

/s/ Bryce Diamant
Bryce Diamant, Book Manager
MSFC/LMSO

/s/ Hideo Sakabe
Hideo Sakabe
NASDA Representative

/s/ Catherine C. Lapenta
Catherine Lapenta
MSFC/FD41

/s/ Alexander Butovchenko
Alexander Butovchenko
RSA Representative

/s/ Lawrence Vezina
Lawrence Vezina
CSA Representative

/s/ Uwe Christ
Uwe Christ
ESA Representative

/s/ Claudio Canu / /s/ Loredana Bruca
Claudio Canu / Loredana Bruca
ASI Representative

INTERNATIONAL SPACE STATION PROGRAM

POIC TO GENERIC USER INTERFACE DEFINITION DOCUMENT, VOLUME 1

REVISION B

JANUARY 2002

APPROVAL

___/s/ Ann McNair_______________
Ann McNair, Manager
MSFC/FD40

REVISION AND HISTORY PAGE

REV.	DESCRIPTION	PUB. DATE
——	Initial Release (Reference per SSCD 003308, EFF. 03/17/00)	05-02-00
A	Revision A (Reference Per SSCD 005648, EFF. 06/14/01)	07-03-01
B	Revision B (Reference per SSCD 006808, EFF. 05/16/02	06-14-02

TABLE OF CONTENTS

PARAGRAPH **PAGE**

APPENDICES

FIGURES

TABLES

EQUATIONS

1.0 INTRODUCTION

1.1 SCOPE

This Interface Definition Document (IDD) describes the standard interfaces between the Marshall Space Flight Center (MSFC) Payload Operations Integration Center (POIC) and a generic user. The term “generic user” is considered synonymous with “remote user”, and the term “remote user” shall be used throughout this document. The following is the definition of a remote user:

> “A remote user is defined as an individual and/or facility, including International Partner Control Centers, who utilizes ground support equipment to access the POIC interfaces. The location of this ground support equipment can either reside at the remote user’s home site, Telescience Support Center (TSC), or in the United States Operations Center (USOC).”

If the user is accessing the POIC using POIC-provided hardware and software from within the USOC, then the user is not a remote user, and the interfaces in this document do not apply. However, if access is from the ground support equipment (GSE) within the USOC, then the user is a remote user.

The ground system interfaces described within this document have been developed to interface with the United States On-orbit Segment (USOS) systems only. The ground systems interfaces described in this document do not interface with the Russian Space Agency (RSA), European Space Agency (ESA), or Japanese Space Agency (NASDA) on-orbit systems.

This IDD contains detailed interface and file format information that will assist the remote user in the development of interface software for GSE. Users seeking comprehensive information on the POIC capabilities are referred to the POIC Capabilities Document (SSP 50304). The POIC Capabilities Document should be considered a complementary document to this document. In cases where an interface is defined, the POIC to Generic User IDD takes precedence. In cases where the capabilities available at the POIC are defined, the POIC Capabilities Document takes precedence. There are also Interface Control Documents (ICD) between specific centers and the POIC. Where these exist, any interface defined in a specific ICD takes precedence over the POIC to Generic User IDD.

The web page https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page/ supplements this document, and provides the following information:

a. Approved SSP 50305 documents and documents in work. This can include the HOSC Management Coordination Group (HMCG) approved version, Ground Segment Control Board (GSCB) approved version, drafts in work, etc.

b. Workarounds due to problems with or partial delivery of POIC capabilities/interfaces.

c. POIC software requirements, such as the supported operating systems and the commercial software required for using POIC X-Windows and web applications.

d. POIC hardware recommendations for interfacing with POIC X-windows and web applications.

e. Operational procedures for interfacing with the POIC.

f. Other pertinent information and notes as needed.

In addition, an e-mail distribution list has been established to provide notification of issues related to remote access of POIC services, most of which are likely to be associated with this document or the POIC Capabilities Document (SSP 50304). To be included on the distribution list, send an e-mail message to joel.best@msfc.nasa.gov.

This IDD is organized as follows:

a. Section 1 provides an introduction to the scope of this document, describes the document control mechanism, and provides an overview of how users can request access to POIC interfaces.

b. Section 2 lists the applicable and reference documentation for the IDD.

c. Section 3 describes the video and voice communications available to remote users, and describes the network protocols utilized by the POIC.

d. Section 4 provides users with information regarding POIC security access requirements and other related security information.

e. Sections 5, 6, and 7 provide users with setup information for the X-Windows, Web, and Enhanced HOSC System Remote Interface Server (ERIS) interfaces, respectively.

f. Section 8 describes telemetry protocols.

g. Section 9 describes the stored telemetry file formats.

h. Section 10 describes information regarding real-time telemetry requests.

i. Section 11 describes command interfaces.

j. Section 12 describes database file formats, including the Ground Support Equipment Definition file format.

k. Section 13 describes file transfer mechanisms for importing and exporting files to and from the POIC.

l. Section 14 describes POIC e-mail protocols and associated information.

m. Section 15 describes Payload Planning System file formats.

n. Section 16 describes information pertaining to the Telescience Resource Kit (TReK) Application Programming Interface (API).

The document text contains To Be Resolved (TBR) identifiers that recognize issues associated with the document contents. The issues are enumerated and are identified by paragraph number in Appendix C. The document text also contains elements not yet defined that are marked with a To Be Determined (TBD) identifier. The TBDs are enumerated and identified by paragraph number in Appendix D.

This IDD is divided into two volumes: Volume 1 describes the standard interfaces as discussed above, and Volume 2 describes security sensitive command uplink and update interfaces. The distribution of Volume 2 will be restricted. In order to obtain Volume 2, contact the NASA Book Manager for SSP 50305, FD41 Mission Systems Development Group at MSFC.

1.2 DOCUMENT CONTROL

This document is controlled by the Ground Segment Control Board (GSCB) as an International Space Station (ISS) Program Office controlled document. All changes to this document will be submitted by the FD41 Book Manager and approved by the GSCB Configuration Change Board. Document changes requested by users should be submitted to the FD41 Book Manager.

1.3 REQUESTING ACCESS TO POIC INTERFACES

The user requests POIC services through the Payload Data Library (PDL). The PDL is available via the Internet. The user may obtain information about a User ID and password for the PDL application by accessing the PDL home page at http://pdl.hosc.msfc.nasa.gov.

From the information provided by the user through the PDL, a POIC account will be established if the user has requested access to any POIC capabilities or interfaces. The POIC's Ground Support Requirements Team (GSRT) will coordinate the establishment of POIC user accounts and determine what software applications are required based on the user’s request. The GSRT will then convey the username, password, and other required information to the user. For more information on the process, see the POIC Capabilities Document (SSP 50304), section 11.

This Page Intentionally Left Blank

2.0 RELATED DOCUMENTATION AND UNIFORM RESOURCE LOCATIONS

2.1 APPLICABLE DOCUMENTS

Applicable documents are documents of the most current issue whose content to the extent specified herein is considered to form a part of this document. The specified documents carry the same weight as if they were stated within the body of this document. The applicable documents are:

CCSDS 701.0-B-2	Recommendation for Space Data Systems Standards: Advanced Orbiting Systems, Networks and Data Links: Architectural Specification, Issue 2.
EIA-170A	Electrical Performance Standards - Monochrome Television Studio Facilities, Jan 1957 (also known as RS-170A)
HOSC-EHS-125	Using the HOSC Computation Generation/Operation Software
HOSC-EHS-136	Using the HOSC Payload Information Management System (PIMS) Software
HOSC-EHS-1124	Using the HOSC Display Generation/Operation Software
HOSC-EHS-1126	Using the HOSC Script Generation/Operation Software
IEEE 802.3	Carrier Sense Multiple Access with Collision Detection Access Method and Physical Layer Specification, 1989.
IEEE 802.3	Information technology--Local and metropolitan area networks--Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, 1993
IEEE 802.3u	Information technology--Local and metropolitan area networks--Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, 1995
ITU-T Recommendation G.703	Physical/electrical characteristics of hierarchical digital interfaces (04/91)

ITU-T Recommendation X.509	Recommendation X.509 - Information technology - Open Systems Interconnection - The directory: Authentication framework (1995), Version 3
JRE 1.1.6	Java Runtime Environment, Version 1.1.6, Javasoft, Sun Microsystems
MIL-STD-188-114A	Electrical Characteristics of Digital Interface Circuits
MSFC-DOC-1949	MSFC HOSC Database Definitions, Volume 4, Telemetry Databases
MSFC-DOC-1949	MSFC HOSC Database Definitions, Volume 5, Command Databases
MSFC-STD-1274	MSFC HOSC Telemetry Format Standard, Volume 2, Packets
RFC 768	Request for Comments: UDP User Datagram Protocol, August 1980
RFC 791	Request for Comments: Internet Protocol, Darpa Internet Program Protocol Specification, September 1981
RFC 792	Request for Comments: Internet Control Message Protocol Darpa Internet Program Protocol Specification, September 1981
RFC 793	Request for Comments: Transmission Control Protocol Darpa Internet Program Protocol Specification, September 1981
RFC 821	Request for Comments: Simple Mail Transfer Protocol, August 1982
RFC 822	Request for Comments: Format of Electronic Mail Messages, August 1982
RFC 854	Request for Comments: Telnet Protocol Specification, May 1983
RFC 855	Request For Comments: Telnet Option Specifications, May 1983
RFC 919	Request For Comments: Broadcasting Internet Datagrams, October 1984

RFC 922	Request For Comments: Broadcasting Internet Datagrams In The Presence Of Subnets, October 1984
RFC 950	Request for Comments: Internet Standard Subnetting Procedure, August 1985
RFC 959	Request for Comments: File Transfer Protocol (FTP), October 1985
RFC 1034	Request for Comments: Domain Names - Concepts and Facilities, November 1987
RFC 1035	Request for Comments: Domain Names - Implementation and Specification, November 1987
RFC 1321	Request for Comments: The MD5 Message-Digest Algorithm, April 1992
RFC 1828	IP Authentication using Keyed MD5, August 1995
SSP-30539	User Interface Language Specification (Timeliner Source)
SSP 41158	Software Interface Control Document, Part 1, United States On-Orbit Segment (USOS) to International Ground System Segment Ku-Band Telemetry Formats
SSP 50304	Payload Operations and Integrations Center (POIC) Capabilities Document
SSP 50401	Multilateral Distributed Planning Interface Specification
X11R6	X Window System Protocol, Version 11 Release 6 by Robert Scheiffler

2.2 REFERENCE DOCUMENTS

Reference documents are those documents that, although not part of this document, serve to amplify or clarify its contents, or dictate work policy or procedures. The specific reference documents are:

MSFC-RQMT-2467	HOSC Information Technology Security Requirements Document
NPG 2810	NASA Procedures and Guidelines: Security of Information Technology

REC-html32	HTML 3.2 Reference Specification, W3C Recommendation by Dave Raggett, January 14, 1997
RFC 1918	Request for Comments: Address Allocation for Private Internets, February 1996
SSP 42018	United States On-orbit Segment (USOS) to Ground through Satellite System (TDRSS) Interface Control Document (ICD)

2.3 APPLICABLE UNIFORM RESOURCE LOCATIONS (URLS)

https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page/index.html

2.4 REFERENCE UNIFORM RESOURCE LOCATIONS

http://trek.msfc.nasa.gov
http://pdl.hosc.msfc.nasa.gov

3.0 COMMUNICATION INTERFACES

This section describes POIC voice communication interfaces, limited POIC video communication interfaces, USOC network communication interfaces, and POIC network protocols. For most U.S. users, including remote user facilities and TSCs, the voice and video information in this section is not applicable. Voice, video, and data interfaces for U.S. remote users and TSCs will be obtained using the user requirements submittal process described in section 11.1 of the POIC Capabilities Document (SSP50304). If the user is performing payload operations in the USOC, sections 3.1.3, 3.2.2, and 3.3.2.1 discuss USOC user GSE interfaces. U.S. users may also be interested in reading section 3.3 to gain insight into the POIC network architecture and its associated network protocols. International Partners will have direct interfaces with the POIC and require the information in this section.

3.1 MISSION/INCREMENT VOICE COMMUNICATION INTERFACES

Mission/increment voice communications are available through multiple interfaces. The interfaces include T-1 (G.703 compliant) and Remote Keyset. The following subsections describe these interfaces.

3.1.1 VOICE T-1 INTERFACE

The POIC provides a T-1 Interface to access required ground-to-ground and air-to-ground voice communications. The T-1 Interfaces support up to 24 embedded voice channels. (See Table 3-1, Electrical Characteristics for T-1.) Voice communications by T-1 are compliant with the G.703 standard defining a 64-Kbps Digital Standard 0 (DS0). Push to Talk (PTT) signaling for air–to-ground support is provided through standard G.703 compliant hardware interfaces.

A user-provided commercially available G.703 Channel Bank or Digital Voice system will provide the demultiplex conversion to strip out the channels. Services in either Alternate Mark Inversion (AMI) or Binary 8 Zero Substitution (B8ZS) encoding and Standard Frame (SF) or Extended Super Frame (ESF) framing are supported.

Table 3-1. Electrical Characteristics for T-1

Cabling	Co-Directional, 120 ohm balanced twisted pair
Framing	Standard Frame (SF), Extended Super Frame (ESF)
Mark	3.0 VDC
Space	0 VDC +/- 0.3 VDC
Pulse Width	647 ns
Encoding	AMI (Bipolar), B8ZS or MIL-STD-188
Speed	1544 Kbps +/- 50 ppm

3.1.2 REMOTE KEYSET

The POIC provides remote voice communications through the 48-button Multiline Phone (MLP). The Remote keyset configuration (see Table 3-2, Electrical Characteristics for Remote keyset DS0) utilizes a user-provided MLP, a user-provided Dual Phone Adapter (DPA), and a user-provided Dual Trunk Adapter (DTA) over either a 56 Kbps or 64 Kbps DS0 circuit. The DS0 must be provided with an Electronic Industries Association (EIA)-530 interface.

Table 3-2. Electrical Characteristics for Remote Keyset DS0

Cabling	Co-Directional, 120 ohm balanced twisted pair
Mark	1.0 VDC
Space	0 VDC +/- 0.3 VDC
Data Rate	64 Kbps
Pulse Width	3.9 usec

3.1.3 USOC VOICE GSE INTERFACE

The USOC provides the capability for the local users to record voice at the console position. Voice Recorders must be user-provided and will interface to the USOC console via a male-to-male RCA cable, which must also be provided by the user.

3.2 VIDEO COMMUNICATIONS

Remote users receive all ISS downlink video from Johnson Space Center (JSC), not from the POIC. For video created within the POIC, the remote user receives the video directly from the POIC. The POIC Communications Facility provides transport and distribution of EIA-170A Composite Video and internal distribution only of limited Red, Green, and Blue (RGB) signals.

3.2.1 COMPOSITE VIDEO

The POIC Communications Facility distributes and displays National Television Standards Committee (NTSC) compliant video streams. (See Table 3-3, Electrical Characteristics for Composite Video.) The internal displayed video is user selectable through local control panels.

Table 3-3. Electrical Characteristics for Composite Video

Electrical Interface	EIA-170A
Nominal Impedance	75 Ohm

3.2.2 USOC VIDEO GSE INTERFACE

The USOC provides the capability for the local users to record video at the console position. Video Recorders must be user-provided and compatible with NTSC video with an EIA-170A interface.

3.3 POIC NETWORK COMMUNICATIONS

This section provides an overview of the POIC network architecture with an emphasis on remote user interfaces. In addition, this section addresses POIC network protocols that are of interest to the remote user and International Partner communities.

3.3.1 POIC NETWORK ARCHITECTURE OVERVIEW

The POIC provides several interfaces for remote users to access POIC data and services. Figure 3-1 identifies the following interfaces to remote users:

- NISN
- International Partner Gateways
- MSFC TSC
- MSFC Institutional Area Network
- User GSE Interface to USOC 10/100 Mbps Switched Ethernet Local Area Networks (LANs)

The NASA Integrated Services Network (NISN) interface is used by the U.S. remote user community, including remote user facilities and TSCs, to access voice, data, and POIC services. The NISN interface also provides Internet access to the POIC. The International Partner Gateways are used by the International Partners to access services described in their Interface Control Documents with the POIC. The MSFC TSC interface provides voice, video, data, and POIC services to users located within the MSFC TSC. The MSFC Institutional Area Network provides local MSFC users with access to the POIC services. Users physically locating GSE in the USOC can connect their GSE to the 10/100 Mbps Ethernet LANs. These LANs provide the user the capability to receive data, access POIC services, and interface with the Internet.

As illustrated in Figure 3-1, the remote user must traverse a Firewall before accessing internal POIC assets. The list of approved protocols and access methods will be covered later in this document.

3.3.2 NETWORK PROTOCOLS

The following sections describe the USOC physical GSE interface and the POIC routing protocols, transport protocols, and standard application protocols. The standards that are associated with each protocol are identified. The status of the OPTIONAL fields that are considered important for remote operations are listed. There are no implications to the support of OPTIONAL fields not mentioned in this document. As for protocol OPTIONS, they will be discussed where appropriate. It is not the intent of this document to explain the inner workings of the protocols used by the HOSC. Should the reader be interested, please refer to the stated standards documentation.

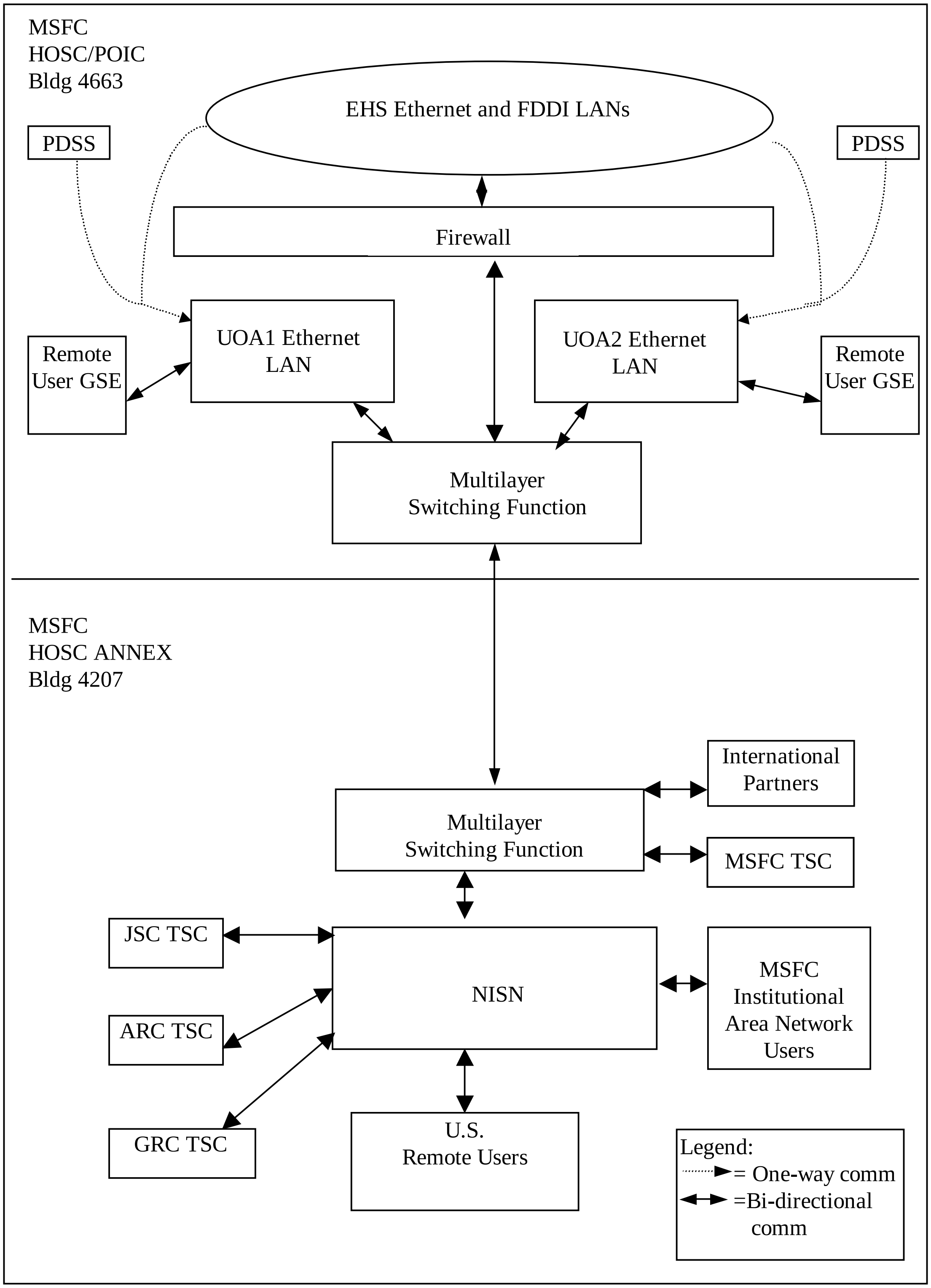


Figure 3-1. POIC Network Architecture Overview

3.3.2.1 USOC PHYSICAL GSE INTERFACE

As shown in Figure 3-1, there are two 10/100 Mbps Ethernet UOA LANs available to support remote user GSE located in the USOC. Each of these UOA LANs provides 48 Ethernet ports with RJ-45 connectors. These LANs will be the source of PDSS telemetry, Custom Data Packet (CDP) telemetry, and GSE packet telemetry. Users will also be allowed to originate connections (File Transfer Protocol (FTP), Telnet, X-Window, etc.) with Internet computers and, if authorized, POIC-provided computers. All connections to POIC computers will be through the POIC Firewall for access to the POIC-provided capabilities. The POIC adheres to the suite of 802.3 Ethernet standards, including 10BASET and 100BASET.

3.3.2.1.1 DELETED

3.3.2.1.2 DELETED

3.3.2.2 ROUTING PROTOCOLS

This section describes the routing protocols used by the POIC. In general, the POIC does not support exchanging routing protocols with user GSE.

3.3.2.2.1 IP

The POIC host implementation of the IP standard adheres to the documented specifications in RFCs 791, 919, 950, and 922. The POIC application software will issue packets capable of being dynamically routed to the final destination.

The POIC will provide a registered IP address for remote user GSE connecting to any of the UOA LANs.

The concept of private IP addresses as documented in RFC 1918 will not be supported. POIC systems ignore the “Type of Service” field, and it is assumed that remote systems and gateways will do the same.

Remote users must use fixed IP addresses on their GSE when accessing the POIC systems. The POIC will also support the use of Class D Multicast IP addresses for remote users that subscribe to the NISN Premium IP network. The POIC will coordinate the multicast IP address assignment upon request.

3.3.2.2.2 ICMP

The hosts and gateways in the POIC support the Internet Control Message Protocol (ICMP) as defined in RFC 792. This protocol is used in various circumstances to aid in the proper operation of the IP network. This protocol is also helpful in troubleshooting network problems and will be used by HOSC Network Managers. The POIC Firewall will not forward ICMP. There is no ICMP support offered to the remote user, except for low level control (i.e., Port/Host unreachable, redirect) messages that should be transparent to the user. The ICMP standard is documented in RFC 792.

3.3.2.3 TRANSPORT PROTOCOLS

3.3.2.3.1 TCP

The Transmission Control Protocol (TCP) is used by HOSC systems. TCP is documented in RFC 793. TCP/IP is the transport protocol used for distribution of CDPs.

3.3.2.3.2 UDP

The User Datagram Protocol (UDP), as documented in RFC 768, is supported and used by POIC systems. UDP is the transport protocol used for the distribution of PDSS telemetry data and GSE packets. POIC systems support the UDP CHECKSUM option field.

3.3.2.4 STANDARD APPLICATION PROTOCOLS

The standard communications application protocols supported by the POIC are defined in the section below. POIC unique application protocols are defined in other sections of this document. For example, POIC telemetry protocols are defined in section 8.0, the POIC ERIS/programmatic interface protocols are defined in section 7.0, and the POIC commanding protocols are defined in Volume 2.

3.3.2.4.1 FILE TRANSFER

The FTP as documented in RFC 959 is supported through the standard TCP port and used by the POIC systems. The remote user has restricted access via FTP as defined in section 13.

3.3.2.4.2 TELNET

The TELNET protocol is NOT allowed through the POIC firewall.

3.3.2.4.3 SMTP/MAIL

The Simple Mail Transfer protocol (SMTP)/Mail access is allowed through the standard TCP port as defined by RFC 821 and 822.

3.3.2.4.4 HTTP

The Hypertext Transfer Protocol (HTTP) is NOT allowed through the POIC Firewall. The POIC allows the remote user to access the POIC Web applications using HTTP access to a Web server located external to the Firewall. See section 6.0 for obtaining access to the Web applications available at the POIC.

3.3.2.4.5 SSH

The Secure Shell (SSH) protocol is a secure packet-based binary protocol that works on top of any transport that will pass a stream of binary data. The SSH protocol is used to access POIC X-Windows capabilities using TCP/IP as the transport. The POIC uses version 1.5 of the SSH Protocol. (See section 5.0 for obtaining X-Windows access to the POIC.)

3.3.2.4.6 SSL3

Secure Socket Layer (SSL3) is used to access POIC Web capabilities using TCP/IP as the transport (See section 6 for obtaining Web access to the POIC). SSL3 uses RC4 encryption with 128-bit encryption key and the MD5 Message Digest Algorithm (RFC 1321). See the subsections below for descriptions of these encryption keys and algorithms. The POIC uses both phases of the SSL Handshake protocol.

> “The SSL (Secure Socket Layer) Handshake Protocol was developed by Netscape Communications Corporation to provide security and privacy over the Internet. The protocol supports server and client authentication. The SSL protocol is application independent, allowing protocols like HTTP FTP (File Transfer Protocol), and Telnet to be layered on top of it transparently. The SSL protocol is able to negotiate encryption keys as well as authenticate the server before data is exchanged by the higher-level application. The SSL protocol maintains the security and integrity of the transmission channel by using encryption, authentication and message authentication codes.”
>
> “The SSL Handshake Protocol consists of two phases, server authentication and client authentication… In the first phase, the server, in response to a client's request, sends its certificate and its cipher preferences. The client then generates a master key, which it encrypts with the server's public key, and transmits the encrypted master key to the server. The server recovers the master key and authenticates itself to the client by returning a message encrypted with the master key. Subsequent data is encrypted with keys derived from this master key. In the second phase, the server sends a challenge to the client. The client authenticates itself to the server by returning the client's digital signature on the challenge, as well as its public-key certificate.”
>
> “A variety of cryptographic algorithms are supported by SSL. During the "handshaking" process, the RSA public-key cryptosystem is used. After the exchange of keys, a number of ciphers are used. These include RC2 RC4, IDEA, DES, and triple-DES. The MD5 message-digest algorithm is also used. The public-key certificates follow the X.509 syntax.”
>
> “The SSL protocol has been submitted as an Internet Draft. Questions can be addressed to <standards@netscape.com>.”

3.3.2.4.6.1 RC4

> “RC4 is a stream cipher designed by Rivest for RSA Data Security. It is a variable key-size stream cipher with byte-oriented operations. The algorithm is based on the use of a random permutation, and analysis shows that the period of the cipher is overwhelmingly likely to be greater than 10100. Eight to sixteen machine operations are required per output byte, and the cipher can be expected to run very quickly in software. While the algorithm is confidential and proprietary to RSA Data Security, Inc., it has been scrutinized under conditions of nondisclosure by independent analysts and is considered secure. The RC4 stream cipher has a special status by which export from the U.S. can often be facilitated.”

3.3.2.4.6.2 MD5 MESSAGE-DIGEST ALGORITHM

MD5 is a message-digest algorithm developed by Rivest. It is meant for digital signature applications where a large message has to be "compressed" in a secure manner before being signed with the private key. The algorithms take a message of arbitrary length and produce a 128-bit message digest. MD5 is aimed at 32-bit machines. Description and source code for the algorithms can be found as Internet RFC 1321.

3.3.2.4.7 CORBA STANDARD

The Common Object Request Broker Architecture (CORBA) is used in the POIC for the Web interface with the remote user for the transfer of files. The CORBA interface is defined using the Internet Inter-ORB Protocol (IIOP) 2 and CORBA Transaction Processing Service 1.1 (CORBA TP1.1). These protocols run on top of the TCP/IP protocol and define the rules and methods for transferring objects between data sources and data recipients. Table 3-5 lists how POIC files are transferred to the generic user. .

Table 3-4. Generic Users File Transfer Methodologies

Data To Be Transferred From POIC	Transfer Method Used
Production Data transferred from PDSS	FTP
Planning/Document files from POIC to Generic User using PIMS to requesting platform	CORBA
Planning/Document files from POIC to Generic User using PIMS to 3rd Party Platform	FTP
Database files from POIC to Generic User	HTTPs

The CORBA is the Object Management Group's answer to the need for interoperability among diverse hardware and software products. Simply stated, CORBA allows applications to communicate with one another no matter where they are located or who has designed them. CORBA 1.1 was introduced in 1991 by Object Management Group (OMG) and defined the Interface Definition Language (IDL) and the API that enable client/server object interaction within a specific implementation of an Object Request Broker (ORB). CORBA 2.0, adopted in December of 1994, defines true interoperability by specifying how ORBs from different vendors can interoperate.

The ORB is the middleware that establishes the client-server relationship between objects. Using an ORB, a client can transparently invoke a method on a server object, which can be on the same machine or across a network. The ORB intercepts the call and is responsible for finding an object that can implement the request, pass it the parameters, invoke its method, and return the results. The client does not have to be aware of where the object is located, its programming language, its operating system, or any other system aspects that are not part of an object's interface. In so doing, the ORB provides interoperability between applications on different machines in heterogeneous distributed environments and seamlessly interconnects multiple object systems.

3.3.2.4.7.1 INTERNET INTER ORB PROTOCOL (IIOP)

The Internet Inter-ORB Protocol (IIOP) specification defines a set of data formatting rules, called Common Data Representation (CDR), which is tailored to the data types supported in the CORBA IDL. Using the CDR data formatting rules, the IIOP specification also defines a set of message types that support the entire ORB semantics defined in the CORBA core specification. Together, the CDR formatting rules and the message formats constitute an abstract protocol called General Inter-ORB Protocol (GIOP). GIOP messages can be sent over virtually any data transport protocol, such as TCP/IP, Novell SPX, SNA protocols, etc. To ensure "out-of-the-box" interoperability between ORB products, the IIOP specification requires that ORBs send GIOP messages over TCP/IP connections because TCP/IP is the standard connection-oriented transport protocol for the Internet. To put it very simply, GIOP TCP/IP = IIOP. IIOP is a mandatory part of CORBA 2.0.

3.3.2.4.7.2 TP 1.1 – TRANSACTION PROCESSING (SERVICE) 1.1

The OMG defines the CORBA Transaction Service as the object-oriented solution for transactions in distributed environments. The ITS Transaction Service includes logging and recovery, and provides the recovery information necessary to ensure that a transaction can always be restored to its rightful state, ensuring consistency between resources, even in the event of system failure.

3.3.2.4.8 INTERNET PROTOCOL SECURITY (IPSEC)

Internet Protocol Security (IPSec) defines a set of standard security protocols that authenticate TCP/IP connections, add data confidentiality and integrity to TCP/IP packets, and are transparent to the application and the underlying network infrastructure. IPSec is designed to support multiple encryption and authentication protocols so a company’s security policy can dictate a desired amount of data privacy and authentication. IPSec enables the idea of secure virtual private networking (VPN), a relatively inexpensive and secure way to connect mobile workers, telecommuters, and branch offices to a corporate site over the Internet or any TCP/IP network. IPSec interoperability enables business partners to communicate securely, even if their network equipment implementations of the standard differ. This could include dedicated security hardware, firewalls, routers, servers and clients, or any network element that implements a TCP/IP stack. Remote users gaining access to the Manual Procedures Viewer (MPV) software will be required to use Checkpoint’s SecuRemote software, which is IPSec-compliant. SecuRemote software requires the following ports to be opened for communication:

TCP/264	-	to retrieve topology downloads
UDP/500	-	IKE (Internet Key Exchange)
IP/50,51	-	encryption of data

3.3.2.5 SUPPORTING NETWORK INFRASTRUCTURE PROTOCOLS

3.3.2.5.1 DOMAIN NAME SERVICE

A Domain Name Service (DNS) is provided by the POIC. The POIC domain is “hosc.msfc.nasa.gov”. The POIC DNS system is RFC 1034 and 1035 compliant. Users located outside the administrative control of the HOSC will be responsible for Internet root DNS service for their hosts if there is a requirement to resolve their host names to IP addresses.

Remote user’s GSE requiring interactive access to the POIC can be added to the HOSC DNS if approved. The POIC Firewall DNS addresses are foehs003.hosc.msfc.nasa.gov (prime) and foehs004.hosc.msfc.nasa.gov (backup).

4.0 SECURITY

This section identifies security procedures or restrictions placed upon the remote user (including International Partner Control Centers) prior to and during access to the POIC. This document only specifies security requirements to access the POIC and ISS. This document does not levy security requirements on the remote user's own site. In a situation where local security rules or regulations are more restrictive, the more restrictive rules and regulations apply. Users are encouraged to employ proactive security measures at their locations over and above what is described herein.

4.1 SENSITIVITY/CRITICALITY

The remote user interface will support telemetry and innocuous and critical commanding. The remote user interface is not authorized to perform any hazardous commanding. See POIC Capabilities Document (SSP 50304), section 4.2, for a description of the various types of commanding capabilities available within the POIC. Authorization for external users to access Mission Systems (MSN) applications is determined on a case-by-case basis where risk analysis has determined that any impact on NASA would NOT be catastrophic, i.e., could NOT result in the loss of major or unique tangible assets and could NOT pose a threat to human life. The determination of the level of risk is agreed upon by the POIC Chief Information Officer (CIO), POIC Computer Security Officer (CSO), and the Data Owner (typically the end user of the application, but needs to be capable of representing all users that could be impacted by the application).

4.2 SECURITY REQUIREMENTS AND POLICIES

POIC security is for the benefit and safety of the ISS and its users. Positive access control is maintained and applicable to all users. The following items support remote user access and connectivity to the POIC services:

1. Access to the POIC shall be protected through the use of the following layered security regimens:
    a. IP address shall be checked at the firewall and web servers as applicable.
    b. Digital certificates shall be used for SSL connections.
    c. SSL connection protocol shall be SSL3, 128-bit encryption, 1024-bit keys, RC4 and MD5 authentication (see section 3.3.2.4.6). (NOTE: International Partners may be delayed in their implementation of web and programmatic interfaces because their government may not allow the import of 128-bit encryption technology.)
    d. Network services will be provided over specified and prearranged ports only.
    e. SSH authentication will be provided with protocol 1.5 (see section 3.3.2.4.5).

2. A digital certificate is issued by machine (remote user GSE) and location (see section 4.5)
    a. Domain Name Service (DNS) Name at the specific site or location, or
    b. Local Name at a specific site or location if a DNS name is not available.

3. Digital Certificates shall be generated and captured through the use of a compliant web browser and access to the POIC certificate services.
    a. These digital certificates shall be used for web and programmatic interfaces.
    b. Digital certificates for programmatic use shall be generated by exporting the platform's operational certificate into the PEM format. The export and generation process is defined on the PGUIDD web interface procedures web page.

4. Accounts are issued by GroupID or UserID
    a. Multiple groups or users may use the same certificate on the same machine
    b. It is allowable to have a second certificate for administrative users

5. Network Address Translation (NAT) is supported with only a one-to-one implementation.

6. Users will be periodically asked to re-authenticate, normally every 24 hours.

7. Only POIC services specifically authorized shall be allowed.

8. Revocation of POIC access can be by GroupID, UserID, or site if the POIC or its mission is threatened.

The POIC implements security in accordance with the HOSC Information Technology Security Requirements (MSFC-RQMT-2467). The remote user security requirements are included in section 4.7, User Site Security Requirements.

4.3 AUTHENTICATION AND AUTHORIZATION DATA

The remote site must take measures necessary to protect the integrity of authentication and authorization data to include passwords, private or secret encryption keys, and digital certificates to access the POIC. Any suspected compromise of authentication and authorization data must be immediately reported to the POIC CSO. Procedures for establishing user accounts and obtaining proper authentication and authorization data are contained in the POIC Capabilities Document (SSP 50304), section 11.

4.4 AUDITING AND MONITORING

The POIC will monitor all events that correspond with access to the POIC. Security relevant events shall be audited and logged. The POIC will work with remote sites to ensure the integrity of the interface. This may at times require that the POIC or NASA use automated scanning software to scan remote systems with interactive access to the POIC for system vulnerabilities. MSFC reserves the right to inspect any site for security risks regardless of where that site is located. For example, a remote user linked to a TSC will be scanned. MSFC will notify all sites in advance when automated scanning will be used at the site.

If it is determined that a remote interface poses a substantial security risk to the POIC, the interface will be terminated. A substantial security risk is defined as any indication of illegal entry into the POIC system or any indication that malicious intent is being performed which may cause damage to the spacecraft or the POIC. This includes network flooding which significantly reduces or stops the productivity of other users. The POIC will work with the remote site to restore connectivity and overall system integrity as soon as possible.

4.5 DIGITAL CERTIFICATE

A POIC digital certificate is a specially formatted digital "passport" that contains information about the user and is signed by the POIC. The digital certificate is used to authenticate the user to the POIC and authorize a web or programmatic connection for further authentication. Each platform has a unique certificate. For programmatic connections, a digital certificate must be exported per the procedure described on the PGUIDD web site, https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page/. This makes the certificate accessible by the user's local application code utilizing SSL3. Users planning to use the CDP or remote commanding interfaces will need to follow these export procedures.

All POIC certificates are in X.509 format that are OSI standards compliant. The POIC will not trust a certificate that is signed by a Certificate Authority other than the POIC. Therefore it is imperative that installation instructions for the digital certificate be followed exactly.

The remote user must use a digital certificate when interfacing with POIC web applications. The remote user must also use a digital certificate when utilizing a programmatic interface. The same digital certificate can be used to access both the web and programmatic interfaces. A digital certificate is not required to access the POIC through X-Windows because that is done through the use of Secure Shell (see section 3.3.2.4.5). Table 4-1 lists the security protocols used for each POIC access method.

Table 4-1. Security Protocols For POIC Access

POIC Access Method	Security Protocol	Certificates/Encryption key Required
X-Window Access (section 5)	SSH	SSH Encryption key provided through COTS
Web Access (section 6)	SSL3	Digital Certificate
Programmatic Access (section 7)	SSL3	Digital Certificate

If you are using the POIC-provided equipment within the USOC to access the POIC web applications via a web browser, then a digital certificate is not required to access the Web browser. However, if you are using a GSE within the USOC, then you are considered a remote user outside the Firewall, and you do need a digital certificate.

The digital certificate received is tied to the remote user’s GSE and the location of the platform that was given upon form completion. The certificate cannot be used on platforms at a different location. The location can either be the domain name service name of the platform or a specific room location if a domain name service name is not available at the remote user site. If the remote user’s GSE becomes unusable, then either:

a. The remote user should have a backup GSE with another personal digital certificate available, or

b. The remote user will have to initialize a replacement platform in the same location of the GSE containing the original digital certificate. This requires that the GSE, including digital certificate information, be backed up properly to allow the user to configure another machine to look exactly like the original machine.

There are many methods to securely store a digital certificate. It is the remote user’s responsibility to keep track of their digital certificate and ensure that their certificate is not stored openly for hackers to access. If help is needed to securely store the information, the user should contact the POIC Help Desk. If at any point a digital certificate is compromised, the POIC can revoke that certificate and another certificate can be generated and installed at the remote user’s site.

To determine if the digital certificate is installed properly for access to web applications, the URL to access the POIC Web Launchpad will start with “https” instead of http to show that the interface to the URL connection is a secure link. This shows the user that they are using secure http protocol (see section 3.3.2.4.4) to access the POIC Web Launchpad (see section 6).

Digital certificates are good throughout the mission/increment to which the remote user is assigned. The user does not have to get another digital certificate or renew the existing digital certificate during that mission/increment. The digital certificate is tied to the platform and its site, not to the explicit Username/platform combination, so if multiple people are sharing a platform at a remote site they do not need to have multiple digital certificates.

The only method that the POIC will offer to get a digital certificate is through the Web browser. This will allow the remote user to have the only copy of the private key that is used by the Public Key Infrastructure (PKI).

Digital certificates give a user a secure link between the POIC and the remote GSE. The connection is now considered private and data can be passed with confidence. This also allows the Java Applets to write on the remote user’s GSE when requested by the user and to allow the remote user to print to their printers.

4.5.1 INITIAL ACQUISITION OF A DIGITAL CERTIFICATE

In order to obtain a digital certificate, the remote user must perform the following steps. See Figure 4-1, Procedure to Acquire A Digital Certificate.

a. A remote user must have a POIC provided UserID and password. The remote user will receive this UserID and Password as a result of the functions performed by the GSRT as identified in section 11 of the POIC Capabilities Document (SSP 50304). This UserID and password will be the same UserID and password used throughout access to the POIC, whether physically located at the POIC, or accessing the POIC remotely through the Web, programmatic, or X-Windows interfaces.

b. In addition to the username and password, the GSRT will give the user a Uniform Resource Locator (URL) to request a digital certificate.

c. Upon entering the specified URL address, the remote user will be transferred to a secure session and will be presented a form to complete. Data requested will be used to verify that the requestor is an authorized certificate owner at a recognized site with the prerequisite platforms identified.

d. Once the form is completed, the user's browser will create a key pair composed of a public and private key. The public key will be sent to the POIC over the encrypted link. The browser will request that the private key be stored in a password-protected file within the browser. It is the user's responsibility to enter a valid password (sometimes called a "pass phrase") to secure the digital certificate. An inadequate password may compromise the certificate. The password will ensure the certificate cannot be exported or used without the owner's permission. Only after this is complete can the remote user terminate the connection and ensure a digital certificate will be created. The POIC will review the online application and notify the remote user of the availability of the digital certificate. Notification will be sent by e-mail and the user will have to connect to a provided URL to receive the digital certificate.

e. The user must install the data exactly as indicated in the instructions that will be included with the digital certificate. The data are sensitive and must be stored according to the following guidelines:

    (1) Cryptographic keys (Data Encrypting Key or Key Encrypting Key) shall be subject to the same security requirements as login passwords.

    (2) The name of a key owner for each cryptographic key shall be recorded in a secure location.

    (3) Systems management shall apply login password requirements to electronically stored cryptographic keys. (See section d above.)

    (4) Systems management shall configure system software to protect cryptographic keys at least equivalent to the protection required for the data being protected.

4.5.2 DELETED

4.6 ACQUIRING SECURE SHELL ENCRYPTION KEY FOR USE AT POIC

The SSH COTS tool provides a capability to create an encryption key while loading the COTS package onto the remote user's GSE. This method will be used to add an encryption key for the SSH into the remote user's GSE.

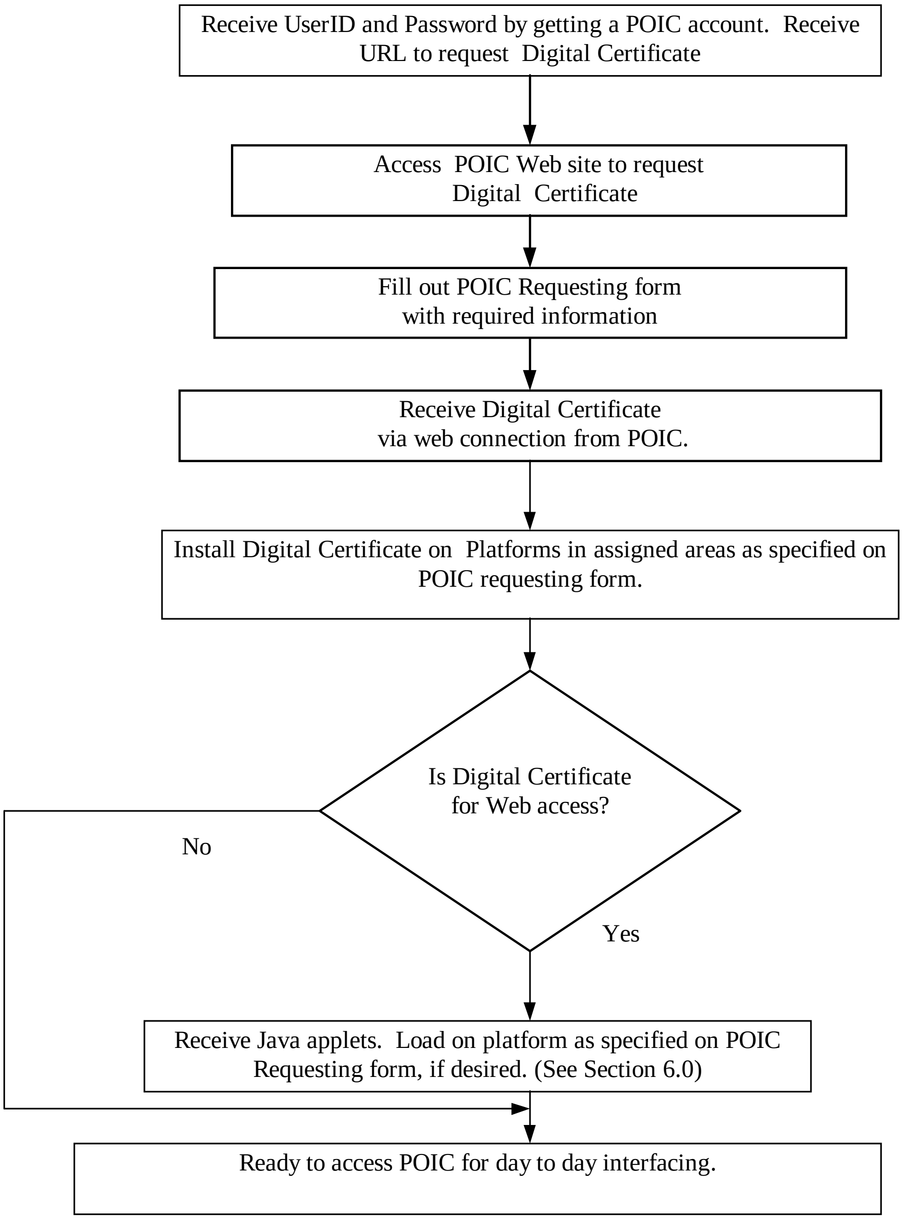

Receive UserID and Password by getting a POIC account. Receive URL to request Digital Certificate
Access POIC Web site to request Digital Certificate
Fill out POIC Requesting form with required information
Receive Digital Certificate via web connection from POIC.
Install Digital Certificate on Platforms in assigned areas as specified on POIC requesting form.
Is Digital Certificate for Web access?
No
Yes
Receive Java applets. Load on platform as specified on POIC Requesting form, if desired. (See Section 6.0)
Ready to access POIC for day to day interfacing.

Figure 4-1. Procedure to Acquire A Digital Certificate

4.7 USER SITE SECURITY REQUIREMENTS

The matrix in Table 4-2 expresses the security requirements levied upon a remote user interacting with the POIC. In defining security requirements there are three different types of remote users, as identified in Table 4-3, Definitions of Users Within Security Requirements. A site shall certify, using the Appendix F security plan format, that adequate security procedures, processes, and programs are in place according to the level of security required (for the type of remote user) to protect POIC and ISS onboard systems from unauthorized access and use.

Table 4-2. User Site Security Requirements

App F Item #	Security Item	Receive Only User	Interactive and Command & Control Users
1	**Security Management** The management process associated with information technology security, such as risk management, auditing, on-going security reviews, risk & vulnerability mitigation, security procedures, agreements, personnel/system/site certification, training, and security oversight. Other more specific security items are listed below.	Not required by NASA Appendix F not required.	A site shall certify that adequate security procedures, processes, and programs are in place commensurate with the level of security required to protect POIC and ISS on-board systems from unauthorized access and use. Site certification shall be based on information provided using Appendix F, Security Plan Checklist for Remote Sites format, and be approved by authorized site management. A site shall report suspected or actual security breaches/incidents to the POIC Computer Security Official (CSO) through the POIC Help Desk. A site shall mitigate security vulnerabilities in a timely manner that may be identified by vulnerability scanning systems or reported through other sources, e.g., security incident report.
2	**Workstation Access Control** Worker/human access to a workstation; e.g., password control.	Not required by NASA	Workstation based POIC related access information (e.g., passwords, encryption keys) shall be protected from unauthorized access.
3	**Configuration Management** The software and hardware configuration of a user's systems.	Not required by NASA	No operational platform shall be configured to allow remote access from unknown or otherwise unauthorized individuals or locations.
4	**Physical Access** Access to the facilities in which payload operations are conducted.	Not required by NASA	Users shall provide control of entry into operational areas.
5	**Network Access and Internet Use** Access by users' systems to local and	Not required by NASA	Sites shall use encryption to protect UserID and password data transmitted between user site and the

Table 4-2. User Site Security Requirements

App F Item #	Security Item	Receive Only User	Interactive and Command & Control Users
	wide area networks.		POIC. Note: Encryption protocols are described in section 3, Communications Interfaces.
6	**Personnel Security** Identification by name of personnel authorized to access POIC systems. Note: No background or personal security checks are required by NASA.	Prior to receipt of telemetry from the POIC, each site shall be authorized in writing by name and appropriate authority to receive telemetry from the POIC.	Each site shall be authorized by name and in writing by appropriate authority for data, voice, and video system access and use. Personnel shall be certified by name to conduct command and control operations.
7	**Audit Trails** A source of information that is not perishable, gathered for the purpose of security management and aiding in criminal investigations.	Not required by NASA	Users shall provide an audit trail of physical access to operational areas. Users shall provide a capability to capture digitized system access data as provided by the system's operating system. A site shall upon request maintain the ability to capture access information associated with a POIC system to include the time of logon and logoff, userID, destination and origination addresses, and the port ID. This information shall be retained for a minimum of 120 days. If the information captured is associated with a reported or suspected security incident, the information shall be retained and protected for an indefinite time.

Table 4-2. User Site Security Requirements

App F Item #	Security Item	Receive Only User	Interactive and Command & Control Users
8	**DB Management System Protection** Protection of user generated data housed in databases.	Not required by NASA	The user shall protect databased command and control information from unauthorized access and use.
9	**Information and Application Protection** Methods to protect systems and data from unauthorized alteration e.g., viruses.	Not required by NASA	Users shall employ, at the site workstation and server level, proactive virus scanning for software/hardware systems that interact with the POIC to ensure that malicious code is not transferred to POIC systems.

Notes on Table 4-2:

- These requirements are for the protection of NASA/POIC assets. Any protection provided by the user in addition to the above security requirements is encouraged.
- Information security items, such as data integrity, availability, and reliability, are not included.
- In the case of conflicting security requirements between a local user/site and these requirements, the more restrictive requirement(s) apply.
- Where a category is “not required by NASA”, NASA encourages a user or site to address those items anyway.

Table 4-3. Definitions of Users Within Security Requirements

Receive Only User:	If a remote user/workstation receives digital streams (telemetry, streaming voice, and/or video) from POIC systems with no other digital interaction, then the remote user is classified as a Receive Only user. For the purpose of user security category determination, voice teleconferencing with POIC cadre personnel that is initiated by the POIC operator is exempt.
Interactive User:	This type of user interacts with the POIC with two-way digital communications over local and wide area networks. An Interactive user/workstation has authority to perform telemetry processing and planning, in addition to the data retrieval. An Interactive user also conducts interactive sessions with POIC systems, including voice and planning video. As a minimum, the requirements for Receive Only users apply to the Interactive User as well.
Command and Control (C&C) User:	In addition to the capabilities of an Interactive user, if a remote user/workstation can initiate payload or experiment command and control, access mission voice A/G and/or has access to planning video, then the remote user is considered a Command and Control (C & C) user.

This Page Intentionally Left Blank

5.0 X-WINDOW SESSION ENVIRONMENT SETUP

This section describes how to set up the remote user's computer for access to the POIC through an X-Windows interface. For information regarding capabilities that are available through the interface, please refer to the POIC Capabilities Document (SSP 50304, section 2.1). The information contained within this section provides a recommendation for the environment setup for X-Windows based on testing performed with the recommended configuration (see section 5.2). Users selecting other configurations are not necessarily guaranteed a successful X-Windows interface.

5.1 X-WINDOWS COMPUTER SOFTWARE CONFIGURATION

To access the POIC through X-Windows, the remote user's GSE must use X-Windows software that is compliant with the following requirements:

a. An X-Windows emulation software package on your system.

b. Remember that with X-Windows the client-server model is "backwards" from normal client-server line of thinking. The remote user's system is considered the X-server side because it is "serving" up the graphics on the remote user's GSE. (See Figure 5-1, X-Windows Client-Server Model.)

c. Able to work with X11R6

d. SSH Client application that is compatible with Secure Shell Protocol version 1.5

e. SSH Encryption key (see Section 4.5.3, Acquiring Secure Shell Encryption Key, for use at POIC)

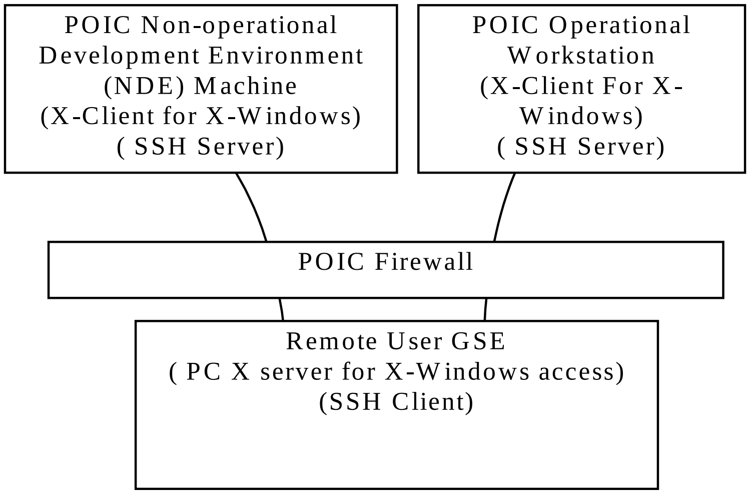

POIC Non-operational Development Environment (NDE) Machine (X-Client for X-Windows) (SSH Server)
POIC Operational Workstation (X-Client For X-Windows) (SSH Server)
POIC Firewall
Remote User GSE (PC X server for X-Windows access) (SSH Client)

Figure 5-1. X-Windows Client-Server Model

The actual COTS software tested for the POIC X-Windows interface is documented at https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page/PGUIDD_Web_Page_3.html. Changes to the X-Windows interface software versions will be coordinated with International Partners prior to making updates to the PGUIDD Web site.

5.2 X-WINDOWS COMPUTER HARDWARE CONFIGURATION

Remote user GSE does not require any special hardware to run POIC X-Windows applications. See https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page/PGUIDD_Web_Page_4.html for the recommended PC and SGI hardware configurations that have been tested and are known to successfully host the POIC X-Windows applications. Changes to the recommended PC and SGI configurations will be coordinated with International Partners prior to making updates to the PGUIDD Web site. Note that the POIC **does *not recommend* the use of Macintosh computers** since this hardware has not been tested and certified with the POIC systems.

5.2.1 DELETED

5.2.2 DELETED

5.3 ACCESSING THE POIC THROUGH X-WINDOWS

The following describes how to set up a computer that is outside the POIC Firewall to access POIC capabilities through X-Windows. The instructions should work independent of the platform used to access the POIC.

5.3.1 PREREQUISITIES FOR LOGIN

The remote user must implement the following prerequisites prior to logging into the POIC:

a. The user must be running an X-Windows emulation software package on their system.

b. The user must install the SSH Client software and SSH encryption key on their system.

c. The user must have an account on the POIC Firewall. This will be provided as part of the POIC account that you get for accessing the X-Windows applications. (See POIC Capabilities Document (SSP 50304), section 11 for obtaining a POIC account.)

d. The user's workstation name and IP address must be resident on the Firewall. This will be done as part of installing the remote user's POIC account into the system. If this causes any problems, the remote user should contact the POIC Help Desk (256-544-5066).

e. The user must have an account on the target system and know the name of the target system. This will be provided as part of the POIC account that you get for accessing the X-Windows applications. (See POIC Capabilities Document (SSP 50304), section 11 for obtaining a POIC account.)

5.3.2 PROCEDURES FOR LOGIN/LOGOUT

For X-Window login/logout procedures, see https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page.

5.3.3 DELETED

6.0 WEB ENVIRONMENT SETUP

This section describes how to set up the user's computer for access to the POIC through a Web browser. For information regarding what capabilities are available through this interface, please refer to the POIC Capabilities Document (SSP50304), section 2.1.

6.1 WEB BROWSER COMPUTER SOFTWARE CONFIGURATION

The PGUIDD Information web site, https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page, provides procedures for installation of all software required to successfully access the POIC web applications. These procedures provide links to the software vendor's web sites where the remote user can download appropriate versions of the required software. Changes to the required web browser versions will be coordinated with the International Partners prior to making updates to the PGUIDD web site.

6.2 WEB BROWSER COMPUTER HARDWARE CONFIGURATION

See https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page/PGUIDD_Web_Page_4.html for recommended platform configurations that have been tested and are known to successfully host the POIC Web applications. Changes to the recommended platform configurations will be coordinated with International Partners prior to making updates to the PGUIDD Web site.

6.2.1 DELETED

6.2.2 DELETED

6.3 DELETED

6.3.1 DELETED

6.3.2 PROCEDURES FOR LOGIN

a. Through the Web browser, the user types in the URL for the POIC external web server. Note that other URLs may be utilized for certain POIC simulations and test activities. These URLs will be provided to the remote user community as needed.

b. Assuming the connection is established, a secure session is established and users receive a login challenge. If the browser is not configured properly, this step will fail even if the user is authorized. (See Figure 6-1, Exchange of Certificate Information, for behind the scenes steps for configuring the secure session.)

c. After entering a UserID and password, the user will be requested to provide MOP information (if required). This allows the Web access to be used in either an operational or off-line support mode depending upon which MOP is chosen.

d. Assuming the user is identified, unique access is allowed for the user. A HOSC Web Launchpad is built dynamically and served to the user's browser automatically. The HOSC Web Launchpad is based on each user's privileges as determined by what the remote user requested (See POIC Capabilities Document (SSP 50304), section 11). A user will not be authorized new privileges without going through a login update request.

e. All users will be re-authenticated every 24 hours. At 23 hours and 50 minutes, the user will be prompted for their password. The user will have 10 minutes and three attempts to enter the password. If re-authentication is not performed or fails, all browser applets are terminated and the browser will be returned to the EHS Web Login screen.

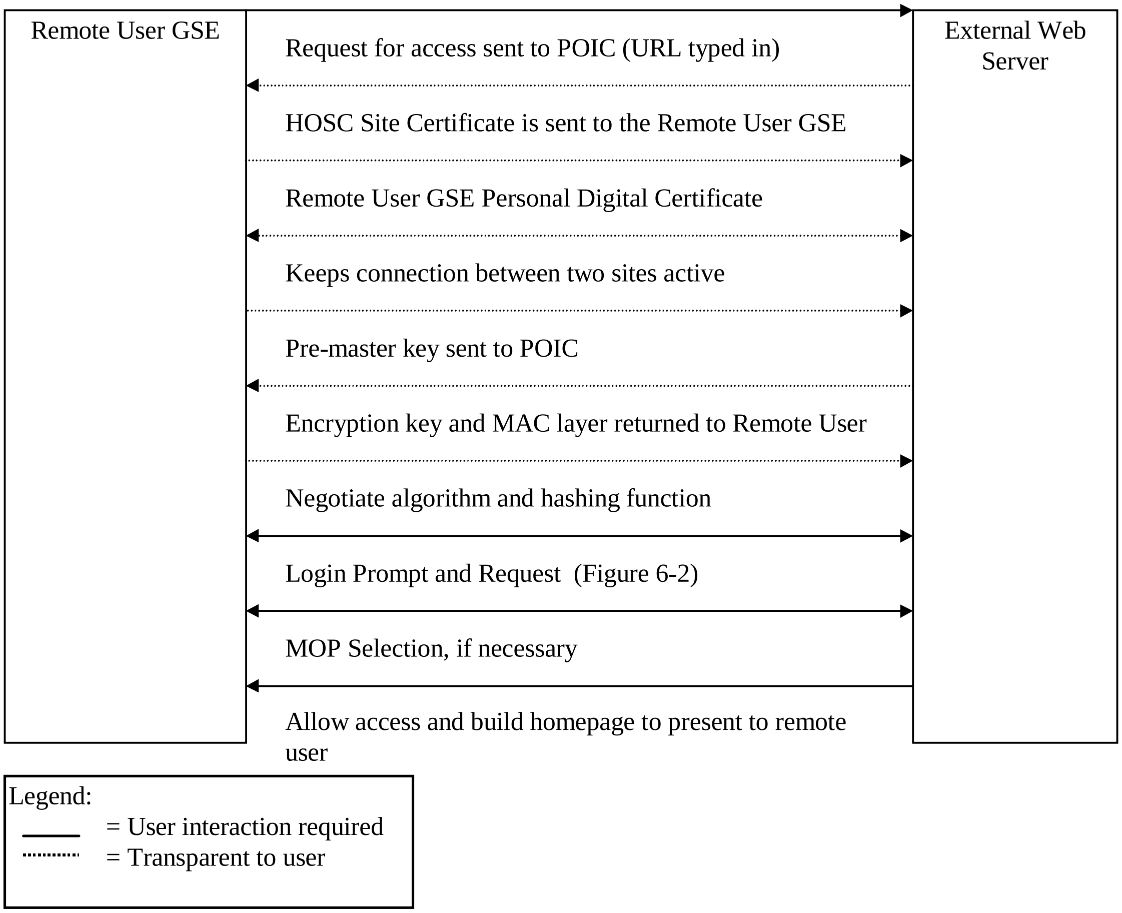

Remote User GSE
External Web Server
Request for access sent to POIC (URL typed in)
HOSC Site Certificate is sent to the Remote User GSE
Remote User GSE Personal Digital Certificate
Keeps connection between two sites active
Pre-master key sent to POIC
Encryption key and MAC layer returned to Remote User
Negotiate algorithm and hashing function
Login Prompt and Request (Figure 6-2)
MOP Selection, if necessary
Allow access and build homepage to present to remote user
Legend:
—— = User interaction required
······ = Transparent to user

Figure 6-1. Exchange of Certificate Information During Login Process

6.3.3 PROCEDURES FOR LOGGING OUT

a. The remote user should use the logoff action under the File option on the web launchpad to terminate web activity with the POIC. This will terminate all applets and provide an orderly shutdown of POIC browser processing. This action will break all web connections to the POIC.

b. The browser on the remote user's GSE will still need to be closed down by the remote user.

7.0 PROGRAMMATIC ENVIRONMENT SETUP

This section contains the detailed definition of the programmatic interface to the POIC, including the interface definition for ERIS and any other programmatic interfaces required to initialize the POIC environment for a programmatic access. For information regarding what programmatic interfaces are available, please refer to the POIC Capabilities Document (SSP 50304), section 4.

A programmatic interface is defined as a direct computer-to-computer where transactions are conducted entirely through software without human intervention. This is in contrast, for example, to a web interface, which requires human intervention to navigate through hyperlinks and download data. If the remote user desires a programmatic interface with the POIC, the software created by the remote user must first interface with the POIC using the ERIS Listener software to initialize the connection with the POIC. Basically, the ERIS Listener establishes a secure connection with the software developed by the remote user and confirms the identity of the remote user. Once that connection is confirmed, the ERIS Listener allows a remote user to initiate a remote user process service request, obtain status, and terminate user process service requests through a programmatic interface.

7.1 PROGRAMMATIC INTERFACE COMPUTER SOFTWARE CONFIGURATION

To access the POIC through a programmatic interface, the user’s ground support equipment must use the following software to access the ERIS:

a. Secure Socket Layer version 3 (SSL3)

b. Use of a POIC issued digital certificate, standard X.509v3 (see section 4.5)

c. RC4 encryption with 128 bit encryption

d. MD5 authentication

7.2 ACCESSING THE POIC THROUGH A PROGRAMMATIC INTERFACE

The following describes how to set up a computer that is outside the POIC Firewall to access POIC capabilities through a programmatic interface. These instructions should work independent of the platform used to access the POIC.

7.2.1 PREREQUISITES FOR LOGIN

The remote user must have the following prerequisites to log into the POIC:

a. You must have SSL3 installed on your system.

b. You must have received and installed a digital certificate based on the export instructions on the PGUIDD Information web site (See section 4.5).

c. You must have an account on the Firewall.

This will be provided as part of the POIC account that you get for requesting programmatic interfaces with the POIC. See POIC Capabilities Document (SSP 50304) section 11, for obtaining a POIC account.

d. You must have an account on the ERIS Server.

This will be provided as part of the POIC account that you get for requesting programmatic interfaces with the POIC.

e. Your system name and IP address must be resident on the Firewall.

This will be done as part of installing the remote user's POIC account into the system. If any problems are encountered, the remote user should contact the POIC Help Desk.

f. For security reasons, any application that establishes outbound socket connections, such as CDP, must have each socket address/port combination pre-approved through the use of a POIC account.

To connect to the ERIS Listener, the software developed by the remote user (remote user process) should establish an encrypted TCP connection to port 9209 on the POIC Firewall using the SSL network API. This connection will be forwarded from the Firewall to the ERIS Listener on an ERIS platform designated ERPVT00n.hosc.msfc.nasa.gov, where n=1, 2, 3, etc.

Once this connection is established between the remote user process and the ERIS Listener, the digital certificate will be exchanged. The exchange of certificates is explained in section 4.5. This will provide initial authentication and will provide a means for the SSL encryption. This connection is the ERIS Listener socket connection.

7.2.2 PROCEDURES FOR LOGIN

After the connection is established and the digital certificate has been properly exchanged, the ERIS Listener will present a username/password challenge to the remote user's process. The ERIS Listener and remote user process will exchange a Username and Password. Upon receiving this message, the ERIS Listener will then attempt to authenticate the user based upon this information and will return one of the Login Status messages.

Upon successful login, the ERIS Listener will attempt to determine which MOP the user's login should be placed. ("MOP" refers to an activity's Mission, Operational Support Mode, and Project, and will be used here to refer to an activity). If the ERIS Listener is unable to retrieve the list of local active MOPs on the ERIS Server or if no MOPs are active, the ERIS Listener will return the MOP List Unavailable Status message or the MOP List Empty Status message respectively and will disconnect the login session.

If there is only one MOP into which the user may login, the MOP Choice Status message will be issued. However, if there are multiple MOPs active into which the user may login, the ERIS Listener will prompt the remote user process for a choice with the MOP Choice Request message. To this prompt, the remote user process should respond with the MOP Response message that indicates the MOP title into which the user desires to login. If a user enters an invalid MOP title, the MOP Choice Range Error Status message will be issued followed by the MOP Choice Request message again. Otherwise the MOP Choice Status message will be issued.

Once a MOP has been determined, the ERIS Listener will establish session information for this login session. An error with this effort will result in a Session Creation Error Status message being sent from the ERIS Listener followed by the login session being disconnected. If no errors occur, however, the ERIS Listener Prompt message will be issued indicating that the ERIS Listener is prepared to receive service requests.

From this point the ERIS Service requests can be started. See Section 7.3, ERIS Service Requests, for further information about continuing the POIC programmatic interface.

7.2.3 PROCEDURES FOR LOGGING OUT

In order to log out of the programmatic interface, all Service Requests should be stopped and an Exit Service Request should be sent to ERIS. Upon receiving the Exit Service Request, any Service Requests that were not stopped by the remote user process will be shut down and terminated. The Exit Service Request will disconnect all TCP/IP socket connections that are still open and disconnect the SSL3 connection.

7.3 ERIS SERVICE REQUESTS

The ERIS Listener supports execution of two types of services: (1) In-Band Services, and (2) Out-of-Band Services.

a. In-Band services return any data through the ERIS Listener socket connection. To avoid intermingling data, only one In-Band request may be executed at a time in a login session. For this same reason, the ERIS Listener will wait on In-Band requests to terminate before issuing another ERIS Listener Prompt message. An example In-Band service is “show_common”, which returns the common configuration information upon request. Since the In-Band service locks out other ERIS Listener requests, the In-Band service is reserved for those service requests that can be handled with an immediate response. Anything else will be handled with an Out-Of-Band Service.

b. To allow multiple services to run simultaneously, ERIS Listener also supports Out-Of-Band services, which return their data through an SSL connection separate from the ERIS Listener socket connection. In preparation for this, the remote user process must be prepared to receive the data on the separate socket. The mechanism for determining how to configure and use this separate connection is dependent on the implementation of each Out-Of-Band service. An example Out-Of-Band service is CDP.

The remote user process can start ERIS services by issuing a Service Request message. Upon receiving a Service Request message, the ERIS Listener will reply with one of the Service Request Status messages. If the ERIS Listener does not recognize the service name, it will return the Unknown Service Status message. If the user does not have the privilege to start the service, the Service Privilege Error Status message will be returned. If the user has started the maximum number of service requests for this login session, the Service Limit Status message will be returned. Problems that the ERIS Listener encounters attempting to execute the service will be noted to the remote user process by the Service Execution Error Status message. If none of these problems occur, the ERIS Listener will respond with either the In-Band Service Started Status message or the Out-Of-Band Service Started Status message, accordingly. Note that the Out-Of-Band Service Started Status message provides a service request ID by which the service may be referenced.

When the ERIS Listener determines that a service has terminated, it will issue one of the Service Termination Status messages. Specifically, if an In-Band Service ended with an error status, then the In-Band Service Error Exit Status message will be issued. If an In-Band service terminated due to receiving an asynchronous notification, the ERIS Listener will issue the In-Band Service Signal Exit Status message. If an Out-Of-Band Service ends for any reason, the Out-Of-Band Service Exit Status message is sent to the remote user process. If the Out-Of-Band Service was terminated due to a Stop Service Request, then the Out-Of-Band Service Exit Status will be returned immediately. If the Out-Of-Band Service terminated other than by a Stop Service Request, the Out-Of-Band Service Exit Status message will be returned just prior to the next ERIS Shell prompt being written in the In-Band communication stream.

7.4 ERIS LISTENER INTERFACES

Due to the transmission of the data across the ERIS socket (i.e., in-band), all data are represented in an ASCII format for user readability. The following tables are specialized to detail the format of each type of data.

7.4.1 EXCHANGING LOGIN PROTOCOLS

The Login Protocols specify the sequence of protocol messages required to establish a Login session with ERIS Listener.

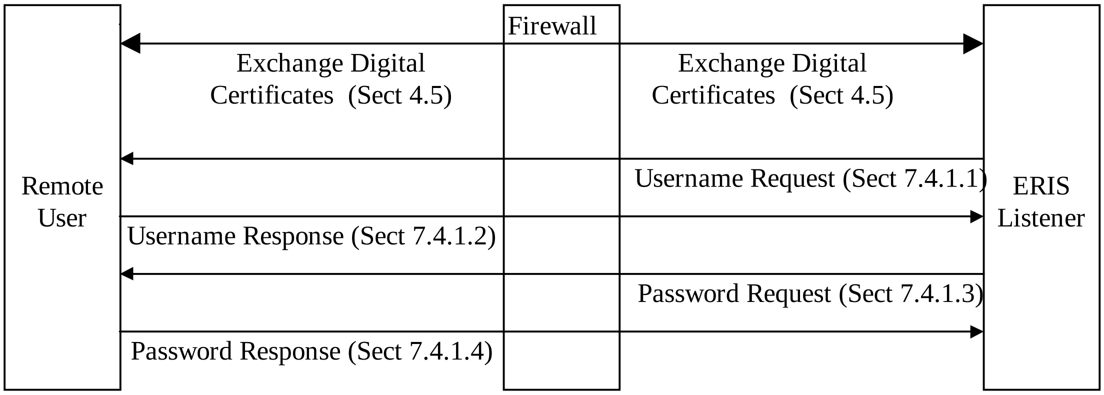

Firewall
Exchange Digital Certificates (Sect 4.5)
Exchange Digital Certificates (Sect 4.5)
Remote User
Username Request (Sect 7.4.1.1)
ERIS Listener
Username Response (Sect 7.4.1.2)
Password Request (Sect 7.4.1.3)
Password Response (Sect 7.4.1.4)

Figure 7-1. Exchanging Login Protocols

7.4.1.1 USERNAME REQUEST

The ERIS Listener will query the remote user process with the Username Request message to initiate the Login session. The remote user process is sent the standard U.S. government disclosure statements prior to receiving the login request. These messages are all given the same message code since parsing is not required when receiving these messages. There are seven lines of this type of message and thus seven message tables of this message code.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-1. Government Disclosure Message 1

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	“0000279035”	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	“ ***”	
14-41	28	White space	Char	<space>	Space added to message.
42-49	8	Government Disclosure Text	Char	“WARNING!”	
50-78	29	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	“***”	
82-83	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:

```
0000279035 ***                         WARNING!                    ***\r\n
```

Table 7-2. Government Disclosure Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	“0000279035”	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	“ ***”	
14-28	15	White space	Char	<space>	Space added to message.
29-63	35	Government Disclosure Text	Char	“This is a U.S. Government computer.”	

Table 7-2. Government Disclosure Message

Byte	Width	Field Name	Type	Values	Description
64-78	15	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	"***"	
82-83	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 ***   This is a U.S. Government computer.   ***\r\n

Table 7-3. Government Disclosure Message 3

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	"0000279035"	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	" ***"	
14-78	65	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	"***"	
82-83	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 ***   ***\r\n

Table 7-4. Government Disclosure Message 4

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	"0000279035"	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	" ***"	
14-15	2	White space	Char	<space>	Space added to message.
16-67	52	Government Disclosure Text	Char	"This system is for the use of authorized users only."	
68-78	11	White space	Char	<space>	Space added to message.

Table 7-4. Government Disclosure Message 4

Byte	Width	Field Name	Type	Values	Description
79-81	3	Asterisk text	Char	“***”	
82-83	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 *** This system is for the use of authorized users only.   *** \r\n

Table 7-5. Government Disclosure Message 5

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	“0000279035”	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	“ ***”	
14-15	2	White space	Char	<space>	Space added to message.
16-76	61	Government Disclosure Text	Char	“By accessing and using the computer system you are consenting”	
77-78	2	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	“***”	
82-83	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 *** By accessing and using the computer system you are consenting ***\r\n

Table 7-6. Government Disclosure Message 6

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	“0000279035”	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	“ ***”	
14-15	2	White space	Char	<space>	Space added to message.

Table 7-6. Government Disclosure Message 6

Byte	Width	Field Name	Type	Values	Description
16-76	61	Government Disclosure Text	Char	"to system monitoring, including the monitoring of keystrokes."	
77-78	2	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	"***"	
82-83	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 *** to system monitoring, including the monitoring of keystrokes. ***\r\n

Table 7-7. Government Disclosure Message 7

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	"0000279035"	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	" ***"	
14-15	2	White space	Char	<space>	Space added to message.
16-74	59	Government Disclosure Text	Char	"Unauthorized use of, or access to, this computer system may"	
75-78	4	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	"***"	
82-83	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 *** Unauthorized use of, or access to, this computer system may   ***\r\n

Table 7-8. Government Disclosure Message 8

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	“0000279035”	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	“ ***”	
14-15	2	White space	Char	<space>	Space added to message.
16-75	59	Government Disclosure Text	Char	“subject you to disciplinary action and criminal prosecution.”	
76-78	3	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	“***”	
82-83	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 *** subject you to disciplinary action and criminal prosecution.   ***\r\n

Table 7-9. Government Disclosure Message 9

Byte	Width	Field Name	Type	Values	Description
0-9	10	Government Disclosure Message Code	Char	“0000279035”	Message code for Government Disclosure Messages
10-13	4	Asterisk text	Char	“ ***”	
14-78	65	White space	Char	<space>	Space added to message.
79-81	3	Asterisk text	Char	“***”	
82-83	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279035 ***   ***\r\n

Table 7-10. Blank Line Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Blank Line Message Code	Char	“0000279001”	Message code for Blank line
10	1	White space	Char	<space>	Space added to message.
11-12	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279001 \r\n

Table 7-11. Username Request

Byte	Width	Field Name	Type	Values	Description
0-9	10	Username Request Message Code	Char	“0000279036”	Message code for Username Request
10	1	White space	Char	<space>	Space added to message.
11-17	7	Username Request	Char	“Login:”	Specifies that the remote user process is required to log into the system.
18-19	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279036 Login:\r\n

All of these messages come at once upon trying to access the ERIS system. Thus the data would look like this on the remote user process interface:

```
0000279035 ***                            WARNING!                              ***\r\n
0000279035 ***               This is a U.S. Government computer.                ***\r\n
0000279035 ***                                                                  ***\r\n
0000279035 ***  This system is for the use of authorized users only.            ***\r\n
0000279035 ***  By accessing and using the computer system you are consenting   ***\r\n
0000279035 ***  to system monitoring, including the monitoring of keystrokes.   ***\r\n
0000279035 ***  Unauthorized use of, or access to, this computer system may     ***\r\n
0000279035 ***  subject you to disciplinary action and criminal prosecution.    ***\r\n
0000279035 ***                                                                  ***\r\n
0000279001 \r\n
0000279036 Login:\r\n
```

7.4.1.2 USERNAME RESPONSE

The remote user process should reply to the Username Request with the Username Response message. This Username is the same UserID that is used on a POIC account for access into the system for Web access or for X-Windows access.

Source:	Remote user process
Destination:	ERIS Listener

Table 7-12. Username Response

Byte	Width	Field Name	Type	Values	Description
0-7	8	Username	Char	Any combination of lowercase alphanumeric characters	Up to 8 characters user name.
8	1	Line Terminator	Char	"\n" (ASCII value 10)	New line character

Example Message:
msl1dmc1\n

7.4.1.3 PASSWORD REQUEST

The ERIS Listener will issue the Password Request message after receiving the Username Response.

Source: ERIS Listener
Destination: Remote user process

Table 7-13. Password Request

Byte	Width	Field Name	Type	Values	Description
0-9	10	Password Request Message Code	Char	"0000279037"	Message code for Password Request
10	1	White space	Char	<space>	Space added to message.
11-18	9	Password Prompt	Char	"Password:"	Specifies that the remote user process is required to enter the password to gain entrance into the system.
19-20	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279037 Password:\r\n

7.4.1.4 PASSWORD RESPONSE

The remote user process should reply with the Password Response message to the Password Request message. The ERIS Listener will then confirm the validity of the Username (UserID) and Password combination. Any responses are discussed in the Login Status section.

Source: Remote user process
Destination: ERIS Listener

Table 7-14. Password Response

Byte	Width	Field Name	Type	Values	Description
0-n-1	1..n	Password	Char	Any combination of non-white characters	Up to n characters password.
n	1	Line Terminator	Char	“\n” (ASCII value 10)	New line character

Example Message:
fsfsd
(Actual password entered will not be seen.)

7.4.2 RECEIVING LOGIN STATUS

Upon receiving the Exchange of Login protocols, the ERIS Listener will then attempt to authenticate the user based upon this information and will return one of the following Login Status messages:

- Invalid Login Status
- Login Privilege Error Status
- Allowed Login Status

If an Allowed Login Status is received, the remote user can still subsequently receive one of the following warning messages, but the login process will continue:

- Login Limit Status
- User Login Limit Status

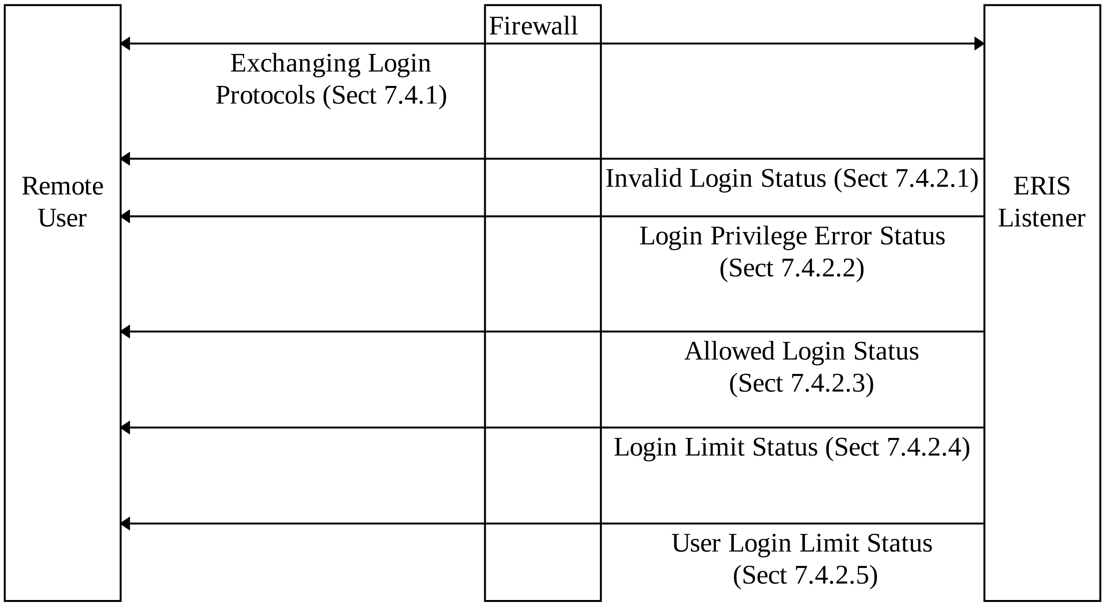

Firewall
Exchanging Login Protocols (Sect 7.4.1)
Remote User
Invalid Login Status (Sect 7.4.2.1)
ERIS Listener
Login Privilege Error Status (Sect 7.4.2.2)
Allowed Login Status (Sect 7.4.2.3)
Login Limit Status (Sect 7.4.2.4)
User Login Limit Status (Sect 7.4.2.5)

Figure 7-2. Receiving Login Status

7.4.2.1 INVALID LOGIN STATUS

If username and password authentication fails, the Invalid Login Status message will be returned. This error message will be given if any of the following conditions occur:

a. The Username (UserID) is not a valid Username for the POIC.

b. The Password is not the correct password for the associated Username

c. Any unprintable characters were inadvertently added to either the Username or Password.

d. The user's account has expired or has been terminated.

e. Note that the Username and Password is case sensitive in Unix. The Username (as specified) is all lowercase.

Source: ERIS Listener
Destination: Remote user process

Table 7-15. Invalid Login Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Incorrect Login Message Code	Char	“0000279038”	Message code for Incorrect Login
10	1	White space	Char	<space>	Space added to message.
11-18	9	Login Incorrect Prompt	Char	“Login incorrect.”	Specifies that the remote user process had an error in the login process.
19-20	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279038 Login incorrect.\r\n

7.4.2.2 LOGIN PRIVILEGE ERROR STATUS

The Login Privilege Error Status message is displayed when an error occurs during the Login process with ERIS Listener. When a remote user receives an account with the POIC, the remote user must have been given the privilege to use the ERIS capability. If this error message is received, then the remote user process did not receive this privilege and the remote user should contact the POIC Help Desk.

Source: ERIS Listener
Destination: Remote user process

Table 7-16. Login Privilege Error Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Login Privilege Error Status Message Code	Char	"0000279030"	Message code for Login Privilege Error
10	1	White space	Char	<space>	Space added to message.
11-30	20	Login Privilege Error Status Text	Char	"Login access denied."	
31-32	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279030 Login access denied. \r\n

7.4.2.3 LOGIN ALLOWED STATUS

The Login Allowed Status message is displayed when a Login process is allowed with ERIS Listener. Subsequently the remote user process may still receive a message on the number of logins allowed, but the remote user process is still allowed into the ERIS Listener. If no subsequent warning messages are received, there are no problems encountered with this login to ERIS Listener.

Source: ERIS Listener
Destination: Remote user process

Table 7-17. Login Allowed Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Login Allowed Status Message Code	Char	"0000279040"	Message code for Login Allowed
10	1	White space	Char	<space>	Space added to message.
11-24	14	Login Allowed Status Text	Char	"Login allowed."	
25-26	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279040 Login allowed. \r\n

7.4.2.4 LOGIN LIMIT STATUS (TBR 7-1)

If the authentication of the remote user process is successful but the maximum number of ERIS Listener logins for the ERIS Server has been reached, the Login Limit Status message will be returned. The maximum number of logins for the ERIS Server is a project configurable number. If this warning message is received, be aware that the resources are being stretched on the ERIS

Server and any services requested during this login session may be rejected due to lack of resources (see section 7.4.6).

Source: ERIS Listener
Destination: Remote user process

Table 7-18. Login Limit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Login Limit Status Message Code	Char	"0000279024"	Message code for Login Limit Status
10	1	White space	Char	<space>	Space added to message.
11-91	81	Login Limit Status Text	Char	"Warning: The recommended maximum number of ERIS login sessions has been exceeded."	
92-93	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279024 Warning: The recommended maximum number of ERIS login sessions has been exceeded.\r\n"

7.4.2.5 USER LOGIN LIMIT STATUS

If the remote user process authentication is successful but the maximum number of ERIS Listener logins for that user has been reached, the User Login Limit Status message will be returned. The maximum number of logins suggested for each Username is also a project configurable number and is the same for all users within that project. As such, if this warning message is received you may want to:

a. Stop a different ERIS login session that you have running on your GSE machine.

b. Ask another team member with the same UserID (if there is one) to stop an ERIS login session that they have running on their GSE machine.

Source: ERIS Listener
Destination: Remote user process

Table 7-19. User Login Limit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	User Login Limit Status Message Code	Char	"0000279031"	Message code for User Login Limit Status
10	1	White space	Char	<space>	Space added to message.
11-83	73	User Login Limit Status Text Part 1	Char	"Warning: The recommended maximum number of ERIS login sessions for user '"	
84-91	8	Username	Char	any combination of lowercase alphanumeric characters	
92-112	21	User Login Limit Status Text Part 2	Char	"' has been exceeded."	
113-114	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279031 Warning: The recommended maximum number of ERIS login sessions for user ' msl1dmc1' has been exceeded.\r\n"

7.4.3 EXCHANGING MOP PROTOCOLS

At this point the ERIS Listener will attempt to determine the MOP where the user's login should be placed. ("MOP" refers to an activity's Mission, Operational, Support Mode, and Project, and will be used here to refer to an activity). The remote user process is either given a choice of which MOP to use or there will only be one MOP available for the user based upon the Username and Password combination presented to ERIS Listener. The remote user process responds to the MOP Choice Request message, if given the option, with the MOP Response. Finally, the MOP Response Status messages give the different status messages that can be received by the remote user process. These are either (1) sending a status of the MOP Choice received, or (2) a situation where the remote user process only has one MOP and only receives notification of the MOP that is associated with the specified login information. Figure 7-3 diagrams the above process.

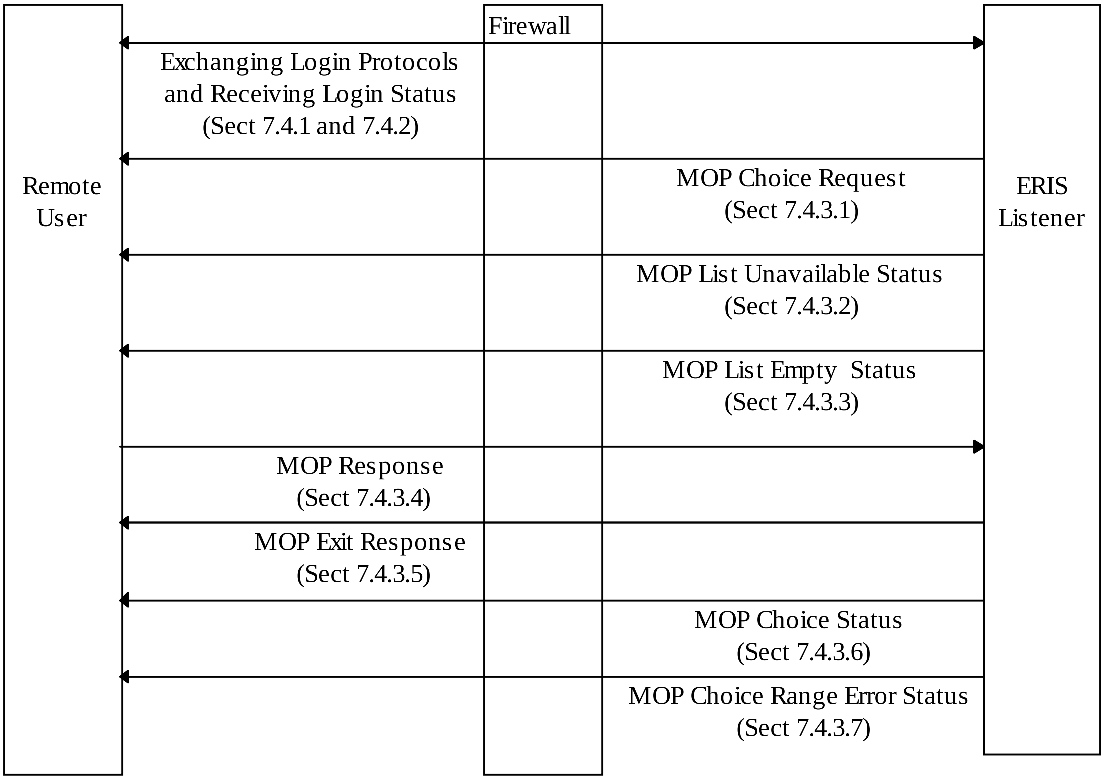

Firewall
Remote User
ERIS Listener
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
MOP Choice Request (Sect 7.4.3.1)
MOP List Unavailable Status (Sect 7.4.3.2)
MOP List Empty Status (Sect 7.4.3.3)
MOP Response (Sect 7.4.3.4)
MOP Exit Response (Sect 7.4.3.5)
MOP Choice Status (Sect 7.4.3.6)
MOP Choice Range Error Status (Sect 7.4.3.7)

Figure 7-3. Exchanging MOP Protocols

7.4.3.1 MOP CHOICE REQUEST

If there are multiple MOPs active into which the user may login, the ERIS Listener will prompt the remote user process with a list of available MOPs and then for the user to enter a choice with the MOP Choice Request message.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-20. Available MOP Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Informational Message Code	Char	“0000279001”	0000279001 is the code for a purely informational message.
10	1	White space	Char	<space>	Space added to message.
11-44	34	MOP Choice Request Text	Char	“Available MOPs into which to login”	
45-46	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279001 Available MOPs into which to login\r\n

The following message will be repeated for every available MOP for the remote user process.

Table 7-21. MOP Available Format

Byte	Width	Field Name	Type	Values	Description
0-9	10	Available MOPS Message Code	Char	"0000279008"	Message code for listing out MOPs that are available to choose from.
10	1	White space	Char	<space>	Space added to message.
11-14	4	MOP Number	Char	0001-9999	Numeric number assigned to MOP preceded by decimal zeros. Lead zeros need not be sent back in reply.
15	1	MOP Delimiter	Char	<:>	MOP Number separator.
16	1	White space	Char	<space>	Space added to message.
17-19*	2..4	Project	Char	"All" "AXAF" "ISS" "SL" "STS"	Allowable project specifications. For Space Station, valid values are "All" and "ISS." Other values may appear on list of MOPs available since ERIS can be a multiproject machine.
20	1	MOP Delimiter	Char	<:>	MOP values separator.
21-24	4	Mission/ Increment	Char	any combination of numeric, uppercase alphabetic, and underscores	4 characters required for Mission or Increment
25	1	MOP Delimiter	Char	<:>	MOP values separator.
26-27 to 26-32	2 -7	Operational Support Mode	Char	"All" "Flight" "Test" "Sim" "VV" "Dev" "Train" "Offline"	Values for different types of Operational Support Modes allowed at POIC.
28-29**	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

*Assuming size based upon ISS entered
**Assuming size based upon smallest entry.
Example Message:
0000279008 0001: ISS:MSL1:Flight\r\n
0000279008 0002: ISS:MSL1:Sim\r\n

Table 7-22. MOP Choice Request Format

Byte	Width	Field Name	Type	Values	Description
0-9	10	MOP Choice Request Message Code	Char	"0000279002"	Message code for MOP Choice Request
10	1	White space	Char	<space>	Space added to message.
11-86	76	MOP Choice Request Text	Char	"Enter the MOP (either number or text) into which to login or 'e' to exit: "	
87-88	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279002 Enter the MOP (either number or text) into which to login or 'e' to exit.:\r\n

These message strings are put together in the following format with blank lines inserted as shown.

Example Message String Sequence:

```
0000279001 \r\n
0000279001 Available MOPs into which to login\r\n
0000279001 \r\n
0000279008 0001: ISS:MSL1:Flight\r\n
0000279008 0002: ISS:MSL1:Sim\r\n
0000279001 \r\n
0000279002 Enter the MOP (either number or text) into which to login or 'e' to exit.: \r\n
```

7.4.3.2 MOP LIST UNAVAILABLE STATUS

If the ERIS Listener is unable to retrieve the list of local active MOPs on the ERIS Server, the ERIS Listener will return the MOP List Unavailable Status message and will disconnect the login session.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-23. MOP List Unavailable Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	MOP List Unavailable Status Message Code	Char	"0000279003"	Message code for MOP List Unavailable
10	1	White space	Char	<space>	Space added to message.
11-88	78	MOP List Unavailable Status Text	Char	"Failed to obtain list of active local MOPs therefore login cannot be accepted."	
89-90	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279003 Failed to obtain list of active local MOPs therefore login cannot be accepted. \r\n

7.4.3.3 MOP LIST EMPTY STATUS

If no MOPs are active, the ERIS Listener will return the MOP List Empty Status message and will disconnect the login session.

Source: ERIS Listener
Destination: Remote user process

Table 7-24. MOP List Empty Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	MOP List Empty Status Message Code	Char	"0000279004"	Message code for MOP List Empty
10	1	White space	Char	<space>	Space added to message.
11-80	70	MOP List Empty Status Text	Char	"No local MOPs are currently active therefore login cannot be accepted."	
81-82	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279004 No local MOPs are currently active therefore login cannot be accepted. \r\n

7.4.3.4 MOP RESPONSE

To the MOP Choice Request prompt, the remote user process should respond with either the MOP Response message or the MOP Exit message (see section 7.4.3.5). The MOP Response message indicates the MOP title into which the user desires to login.

Source: Remote user process
Destination: ERIS Listener

Table 7-25. MOP Response

Byte	Width	Field Name	Type	Values	Description
0-2*	2..4	Project	Char	“All” “AXAF” “ISS” “SL” “STS”	Allowable project specifications. For Space Station, valid values are “All” and “ISS.” Other values may appear on list of MOPs available since ERIS can be a multiproject machine.
3	1	MOP Delimiter	Char	<:>	MOP values separator.
4-7	4	Mission/ Increment	Char	Any combination of numeric, uppercase alphabetic, and underscores	4 characters required for Mission or Increment
8	1	MOP Delimiter	Char	<:>	MOP values separator.
9-10 to 9-15	2 -7	Operational Support Mode	Char	“All” “Flight” “Test” “Sim” “VV” “Dev” “Train” “Offline”	Values for different types of Operational Support Modes allowed at POIC.
11- 16	1	Line Terminator	Char	“\n” (ASCII value 10)	New line character

***Size based on ISS project.**
Example Message:
ISS:MSL1:Flight\n

7.4.3.5 MOP EXIT RESPONSE

To the MOP Choice Request prompt, the remote user process can respond with either the MOP Response message (see section 7.4.3.4) or the MOP Exit. The MOP Exit Response will exit the remote user process from the ERIS Listener.

Source: Remote user process
Destination: ERIS Listener

Table 7-26. MOP Exit Response

Byte	Width	Field Name	Type	Values	Description
0	1	Exit information	Char	“e”	Response after MOP Choice message to exit from ERIS Listener
1	1	Line Terminator	Char	“\n” (ASCII value 10)	New line character

Example Message:
e\n

7.4.3.6 MOP CHOICE STATUS

The MOP Choice Status message is sent back to the remote user process to confirm the MOP that was chosen. If there is only one MOP into which the user may login, the MOP Choice Status message will be issued.

Source: ERIS Listener
Destination: Remote user process

Table 7-27. MOP Choice Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	MOP Choice Status Message Code	Char	“0000279006”	Message code for MOP Choice Status Message used to verify the MOP chosen by the remote user process.
10	1	White space	Char	<space>	Space added to message.
11-16	6	MOP Choice Status Text Part 1	Char	“User '”	
17-24	8	Username	Char	Any combination of lowercase alphanumeric characters	
25-50	26	MOP Choice Status Text Part 2		“' will be working in MOP '”	Middle part of message including starting quote for MOP value.
51-53	3	Project	Char	“All” “AXAF” “ISS” “SL” “STS”	Allowable project specifications. For Space Station, valid values are “All” and “ISS”.
54	1	MOP Delimiter	Char	<:>	MOP values separator.
55-58	4	Mission/ Increment	Char	any combination of numeric, uppercase alphabetic, and underscores	4 characters required for Mission or Increment

Table 7-27. MOP Choice Status

Byte	Width	Field Name	Type	Values	Description
59	1	MOP Delimiter	Char	<:>	MOP values separator.
60-61 to 60-66	2 -7	Operational Support Mode	Char	“All” “Flight” “Test” “Sim” “VV” “Dev” “Train” “Offline”	Values for different types of Operational Support Modes allowed at POIC.
62-63 to 67-68	2	MOP Choice Status Text Part 3		“'.”	Sentence is finished with the second quote and the period.
64-65 to 69-70	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279006 User 'msl1dmc1' will be working in MOP 'ISS:MSL1:Flight’. \r\n

When this message is sent to the remote user process it is surrounded by two blank lines to make sure it is easy to read. Thus the actual message will appear as:

0000279001 \r\n
0000279006 User 'msl1dmc1' will be working in MOP 'ISS:MSL1:Flight’. \r\n
0000279001 \r\n

7.4.3.7 MOP CHOICE RANGE ERROR STATUS

If a user enters an invalid MOP title, the MOP Choice Range Error Status message will be issued followed by the MOP Choice Request message again. Otherwise the MOP Choice Status message will be issued.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-28. MOP Choice Range Error Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	MOP Choice Range Error Status Message Code	Char	"0000279005"	Message error code for MOP Choice Message used to specify that the MOP value was not in range.
10	1	White space	Char	<space>	Space added to message.
11-80	70	MOP Choice Range Error Status Text Part 1	Char	"MOP choice must be a listed MOP title, 'e', or a number between 1 and "	
81 to 84	1..4	MOP Count	Char	"1".. "9999"	
82-94 to 85-97	13	MOP Choice Range Error Status Text Part 2		". Try again."	Final part of message.
95-96 to 98-99	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279005 MOP choice must be a listed MOP title or a number between 1 and 45. Try again. \r\n

7.4.4 RECEIVE ERIS LISTENER ENTRY

Once a MOP has been determined, ERIS Listener will establish session information for this login session. If no errors occur, the ERIS Listener Prompt message will be issued indicating that the ERIS Listener is prepared to receive service requests.

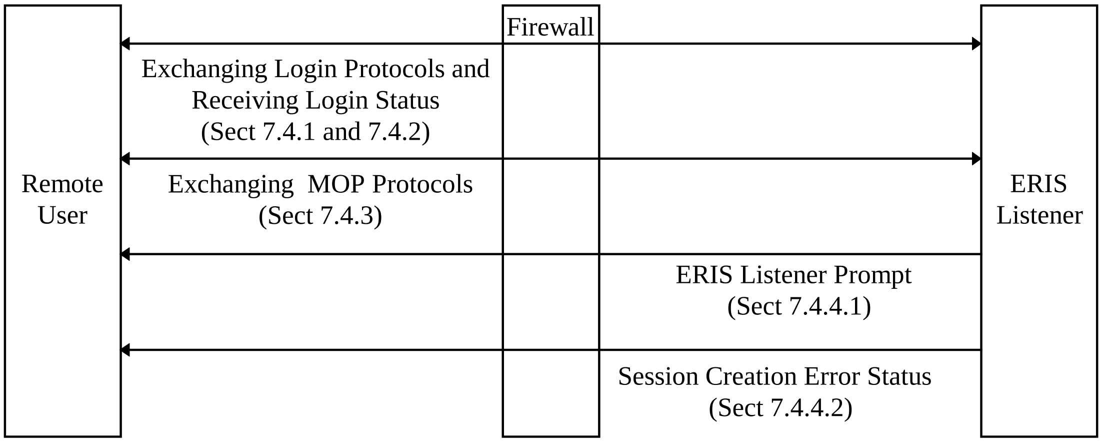

Firewall
Remote User
ERIS Listener
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
Exchanging MOP Protocols (Sect 7.4.3)
ERIS Listener Prompt (Sect 7.4.4.1)
Session Creation Error Status (Sect 7.4.4.2)

Figure 7-4. Receive ERIS Listener Entry

7.4.4.1 ERIS LISTENER PROMPT

If no errors occurred during the Login process, the ERIS Listener Prompt message will be issued indicating that the ERIS Listener is prepared to receive service requests. The user can respond with any of the User initiated responses at this time. It should also be noted that any time an ERIS Listener Prompt message is displayed the out-of-band communications are checked to determine if any termination messages need to be included in the In-Band communication stream.

Source: ERIS Listener
Destination: Remote user process

Table 7-29. ERIS Listener Prompt

Byte	Width	Field Name	Type	Values	Description
0-9	10	ERIS Listener Prompt Message Code	Char	“0000279000”	Message code for ERIS Listener Prompt
10	1	White space	Char	<space>	Space added to message.
11-23	13	ERIS Listener Prompt Text	Char	“eris_shell > ”	Remote user process can respond with request after receiving this prompt.
24-25	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279000 eris_shell >\r\n

7.4.4.2 SESSION CREATION ERROR STATUS

If an error occurs while ERIS Listener is establishing session information for this login session, the ERIS Listener will send a Session Creation Error Status message and the login session will be disconnected.

Table 7-30. Session Creation Error Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Session Creation Error Status Message Code	Char	“0000279007”	Message code for Session Creation Message used to verify an error occurred during creation of session.
10	1	White space	Char	<space>	Space added to message.
11-52	42	Session Creation Error Status Error Status Text Part 1	Char	“Error: Could not create user's session -- ”	Initial message

Table 7-30. Session Creation Error Status

Byte	Width	Field Name	Type	Values	Description
53+ n-1	1..n	Error Reason	Char	Any combination of printable characters	Operating System error received upon trying to create session.
53+n	1	Session Creation Error Status Error Status Text Part 2		"."	Final text portion of message.
54+n - 55+n	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279007 Error: Could not create user's session -- <Error message from System returned here>.\r\n

7.4.5 SERVICE REQUEST TYPES

The ERIS Listener supports execution of two types of services: (1) In-Band Services and (2) Out-Of-Band Services. In-Band services return any data through the ERIS Listener socket connection. To avoid intermingling data, only one In-Band request may be executed at a time in a login session. For this same reason, the ERIS Listener will wait on In-Band requests to terminate before issuing another ERIS Listener Prompt message.

To allow multiple services to run simultaneously, ERIS Listener also supports Out-Of-Band services, which return their data through a SSL connection separate from the ERIS Listener socket connection. In preparation for this, the remote user process must be prepared to receive the data on the separate socket. The mechanism for determining how to configure and use this separate connection is dependent on the implementation of each Out-Of-Band service.

The general form of a service request is indicated in the following table. However, tables are also included for specific ERIS services in following sections.

Table 7-31. Service Request

Byte	Width	Field Name	Type	Values	Description
0-n-1	1 ..n	Service Name	Char	Any combination of non-white space characters.	Desired service by the remote user process. The names of the services allowed into ERIS are discussed in the following sections.
n	1	White space	Char	<tab>, <space>	Space added to message.
n+1-n+p	1 ..p	Service Arguments	Char	Any combination of words separated by white space.	Any arguments provided for the service name can be separated by spaces or tabs.
n+p +1	1	Line Terminator	Char	"\n" (ASCII value 10)	New line character

Note: Figure 7-5 shows the interaction between the In-Band Service Requests and the ERIS status messages. This figure starts with the ERIS Shell prompt being available.

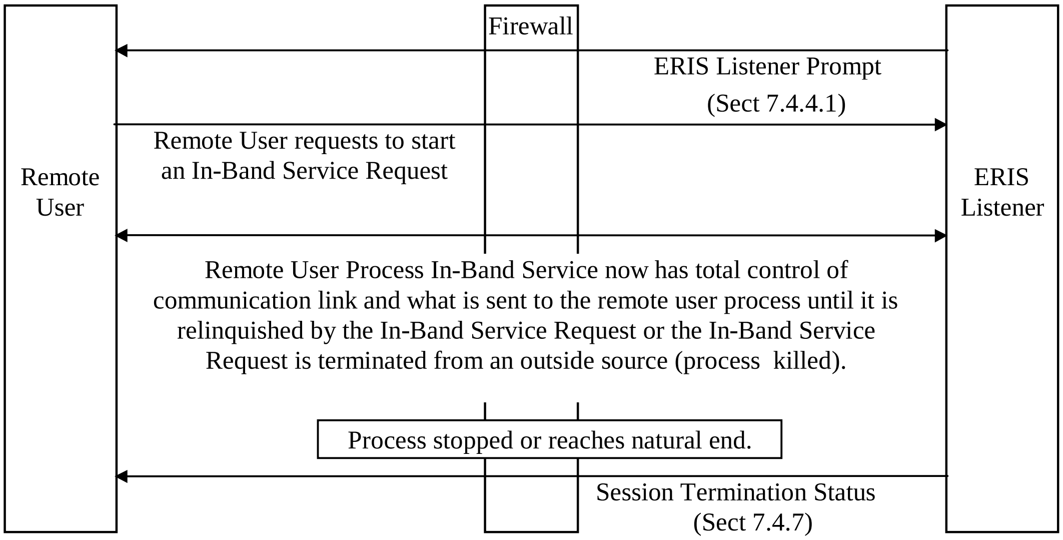

Remote User
Firewall
ERIS Listener
ERIS Listener Prompt
(Sect 7.4.4.1)
Remote User requests to start an In-Band Service Request
Remote User Process In-Band Service now has total control of communication link and what is sent to the remote user process until it is relinquished by the In-Band Service Request or the In-Band Service Request is terminated from an outside source (process killed).
Process stopped or reaches natural end.
Session Termination Status
(Sect 7.4.7)

Figure 7-5. In-Band Service Request Process

7.4.6 SERVICE REQUEST STATUS

Upon receiving a Service Request message, the ERIS Listener will reply with one of the following Service Request Status messages. The following are status messages issued upon the initiation of a Service Request and can be returned to the remote user process by the ERIS Listener for any type of Service Request. Based upon the type of Service Request (see Table 7-32 for the list of the sections that contain the Service Requests), there will also be the appropriate Service Request type response.

Table 7-32. POIC Service Request Types

Service Request Name	Request Type	Section
Status Request	In-Band	7.5
Stop Request	In-Band	7.6
Exit Request	In-Band	7.7
Common Configuration Request	In-Band	7.8
Custom Data Packet Request	Out-of-Band	7.9
Remote Command Service Request	Out-Of-Band	7.10

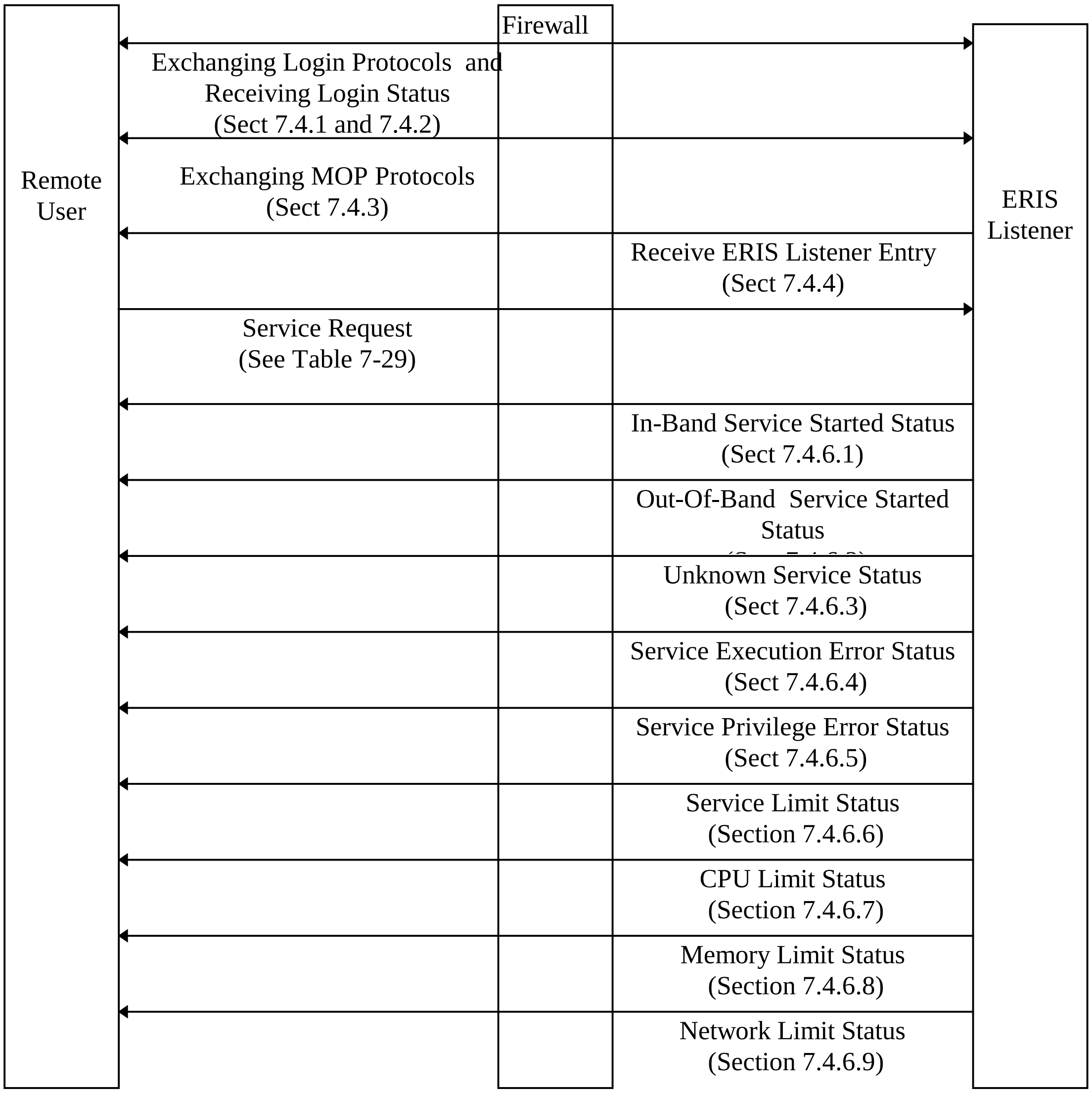

Firewall
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
Remote User
Exchanging MOP Protocols (Sect 7.4.3)
ERIS Listener
Receive ERIS Listener Entry (Sect 7.4.4)
Service Request (See Table 7-29)
In-Band Service Started Status (Sect 7.4.6.1)
Out-Of-Band Service Started Status
Unknown Service Status (Sect 7.4.6.3)
Service Execution Error Status (Sect 7.4.6.4)
Service Privilege Error Status (Sect 7.4.6.5)
Service Limit Status (Section 7.4.6.6)
CPU Limit Status (Section 7.4.6.7)
Memory Limit Status (Section 7.4.6.8)
Network Limit Status (Section 7.4.6.9)

Figure 7-6. Service Request Status Responses

7.4.6.1 IN-BAND SERVICE STARTED STATUS

If no problems occur, the ERIS Listener will respond with the In-Band Service Started Status message to an In-Band Request Message.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-33. In-Band Service Started Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	In-Band Service Started Status Message Code	Char	“0000279020”	Message code for specifying that an In-Band service was started by the remote user process.
10	1	White space	Char	<space>	Space added to message.
11-19	9	In-Band Service Started Status Text Part 1	Char	“Service '”	Initial text for message
20 +n-1	1..n	Service Name	Char	any combination of non-white space characters	
20+n	1	White space	Char	<space>	Space added to message.
20+ n+p-1	1..p	Service Arguments	Char	any combination of words separated by white space	
30+ n+p-1	10	In-Band Service Started Status Text Part 2	Char	“' started.”	
30+ n+p - 31 +n+p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279020 Service ‘show_common' started. \r\n

7.4.6.2 OUT-OF-BAND SERVICE STARTED STATUS

If no problems occur, the ERIS Listener will respond with the Out-Of-Band Service Started Status message to the Out-Of-Band Service Request Message. Note that the Out-Of-Band Service Started Status message provides a service request ID by which the service may be referenced.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-34. Out-Of-Band Service Started Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Out-Of-Band Service Started Status Message Code	Char	“0000279021”	Message code for specifying that an Out-Of-Band service was started by the remote user process.
10	1	White space	Char	<space>	Space added to message.
11-27	17	Out-Of-Band Service Started Status Text Part 1	Char	“Service request '”	
28-29*	1..2	Service Request ID		“1” .. max	Maximum value <= 99.
29* - 40	12	Out-Of-Band Service Started Status Text Part 2	Char	“’ (service ‘”	
40* to 40+n-1	1..n	Service Name	Char	any combination of non-white space characters	
40*+n	1	White space	Char	<space>	Space added to message.
41*+n to 41*+n+p-1	1..p	Service Arguments	Char	any combination of words separated by white space	
41*+n+p to 51+n+p	11	Out-Of-Band Service Started Status Text Part 3		“’) started."	
52+n+p-53+n+p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279021 Service request ‘12’ (service ‘CDP 111.222.123.234 1234') started. \r\n

7.4.6.3 UNKNOWN SERVICE STATUS

If the ERIS Listener does not recognize the service name, it will return the Unknown Service Status message.

Source: ERIS Listener
Destination: Remote user process

Table 7-35. Unknown Service Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Unknown Service Status Message Code	Char	Set to “0000279009”	Message code for specifying that the service requested is an unknown type.
10	1	White space	Char	<space>	Space added to message.
11-19	19	Unknown Service Status Text Part 1	Char	“Requested service '”	
20 - 20+n -1	1..n	Service Name	Char	any combination of non-white space characters	
20+n to 32+n	13	Unknown Service Status Text Part 2	Char	“' is unknown.”	
33+n - 34+n	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279009 Requested service 'show_commmon' is unknown. \r\n

7.4.6.4 SERVICE EXECUTION ERROR STATUS

Problems that the ERIS Listener encounters attempting to execute the service will be noted to the remote user process by the Service Execution Error Status message. The Service Execution Error Reason that is returned with this message is dependent upon the error message that was received from the UNIX system that created the error message.

Source: ERIS Listener
Destination: Remote user process

Table 7-36. Service Execution Error Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Service Execution Error Status Message Code	Char	“0000279010”	Message code for specifying that a service had an execution error upon starting.
10	1	White space	Char	<space>	Space added to message.
11-35	25	Service Execution Error Status Text Part 1	Char	“Could not start service '”	
36 to 36+n-1	1..n	Service Name	Char	Any combination of non-white space characters	

Table 7-36. Service Execution Error Status

Byte	Width	Field Name	Type	Values	Description
36+n	1	White space	Char	<space>	Space added to message.
36+n to 36+n+p-1	1..p	Service Arguments	Char	Any combination of words separated by white space	
(36+n+p) to (47+n+p)	12	Service Execution Error Status Text Part 2	Char	“', because '”	
(48+n+p) to (48+n+p +r-1)	1..r	Error Reason		Any combination of printable characters	
(48+n+p +r) to (49+n+p +r)	2	Service Execution Error Status Text Part 3		“'.”	
50+n+p+r - 51+n+p+r	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279010 Could not start service 'show_common', because 'ERROR MESSAGE FROM SYSTEM PRINTED HERE.’ \r\n

7.4.6.5 SERVICE PRIVILEGE ERROR STATUS

If the user does not have privilege to start the service, the Service Privilege Error Status message will be returned.

Source: ERIS Listener
Destination: Remote user process

Table 7-37. Service Privilege Error Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Service Privilege Error Status Message Code	Char	“0000279022”	Message code for specifying that a service had a privilege error upon starting.
10	1	White space	Char	<space>	Space added to message.
11-19	9	Service Privilege Error Status Text Part 1	Char	“Service '”	

Table 7-37. Service Privilege Error Status

Byte	Width	Field Name	Type	Values	Description
20 to 20+n -1	1..n	Service Name	Char	Any combination of non-white space characters	
20+n to 61+n	42	Service Privilege Error Status Text Part 2	Char	“' denied: user not privileged for service.”	
62+n - 63+n	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279022 Service 'show_common' denied: user not privileged for service. \r\n

7.4.6.6 SERVICE LIMIT STATUS

If the user has started the maximum recommended number of service requests for this login session, the Service Limit Status message will be returned. However, the service request will still be started if the Central Processing Unit (CPU), Memory, and Network limits have not been surpassed.

Source: ERIS Listener
Destination: Remote user process

Table 7-38. Service Limit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Service Limit Status Message Code	Char	“0000279023”	Message code for specifying that a service exceeded the limited number of services allowed upon starting.
10	1	White space	Char	<space>	Space added to message.
11-94	84	Service Limit Status Text Part 1	Char	“Warning: The recommended maximum number of user service requests has been exceeded.”	
95-96	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279023 Warning: The recommended maximum number of user service requests has been exceeded.\r\n

7.4.6.7 CPU LIMIT STATUS

The ERIS listener process will check when a remote user process is being accepted to determine if there is sufficient Central Processing Unit (CPU) for another process to be started. If the CPU Limitation currently configured on the ERIS Server has been reached, then the CPU Limit status will be returned to the remote user process. The remote user can wait until later to resubmit the process or contact the POIC Help Desk.

Source: ERIS Listener
Destination: Remote user process

Table 7-39. CPU Limit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	CPU Limit Status Message Code	Char	“0000279032”	Message code for CPU Limit Status
10	1	White space	Char	<space>	Space added to message.
11-19	9	CPU Limit Status Text Part 1	Char	“Service '”	
20 +n-1	1..n	Service Name	Char	Any combination of non-white space characters	
20+n to 68+n	48	CPU Limit Status Text Part 2	Char	“' denied: CPU utilization is currently too high.”	
69+n to 70+n	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
“0000279032 Service 'CDP' denied: CPU utilization is currently too high.\r\n”

7.4.6.8 MEMORY LIMIT STATUS

The ERIS listener process will check when a remote user process is being accepted to determine if there is sufficient memory available in the ERIS Server for another process to be started. If the Memory Limitation currently configured on the ERIS Server has been reached, then the Memory Limit status will be returned to the remote user process. The remote user can wait until later to resubmit the process or contact the POIC Help Desk.

Source: ERIS Listener
Destination: Remote user process

Table 7-40. Memory Limit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Memory Limit Status Message Code	Char	"0000279033"	Message code for Memory Limit Status
10	1	White space	Char	<space>	Space added to message.
11-19	9	Memory Limit Status Text Part 1	Char	"Service '"	
20+n-1	1..n	Service Name	Char	Any combination of non-white space characters	
20+n to 71+n	51	Memory Limit Status Text Part 2	Char	"' denied: Memory utilization is currently too high."	
72+n to 73+n	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279033 Service 'CDP' denied: Memory utilization is currently too high.\r\n

7.4.6.9 NETWORK LIMIT STATUS

The ERIS listener process will check when a remote user process is being accepted to determine if there is sufficient networking bandwidth for another process to be started. If the Network Limitation currently configured on the ERIS Server has been reached, then the Network Limit status will be returned to the remote user process. The remote user can wait until later to resubmit the process or contact the POIC Help Desk.

Source: ERIS Listener
Destination: Remote user process

Table 7-41. Network Limit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Network Limit Status Message Code	Char	"0000279034"	Message code for Network Limit Status
10	1	White space	Char	<space>	Space added to message.
11-19	9	Network Limit Status Text Part 1	Char	"Service '"	
20 +n-1	1..n	Service Name	Char	Any combination of non-white space characters	
20+n to 72+n	52	Network Limit Status Text Part 2	Char	"' denied: Network utilization is currently too high."	

Table 7-41. Network Limit Status

Byte	Width	Field Name	Type	Values	Description
73+n to 74+n	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279034 Service 'CDP' denied: Network utilization is currently too high.\r\n

7.4.7 SERVICE TERMINATION STATUS

When the ERIS Listener determines that a service has terminated, it will issue one of the Service Termination Status messages. There is no Stop or Exit Service Request for an In-Band Request. By definition, an In-Band request has control of the communication link, so another request cannot be started (i.e., a Stop Request). However, when the In-Band Service Request is finished, the In-Band Service Status messages will let the remote user process know if the exit was a normal exit or if an error occurred which caused the exit. When an Exit Service Request is received, no In-Band or Out-of-Band Exit Status messages are posted (see Section 7.7, Exit Service Request). Before any ERIS Shell prompt is displayed to the user, the ERIS Listener checks to determine if any service requests have been terminated. If an Out-Of-Band Service Exit Status has not been displayed to the remote user process and the information is available, then the status message will appear at that time. There will be one Out-Of-Band Service Exit Status message for every Service Request that was found to have terminated.

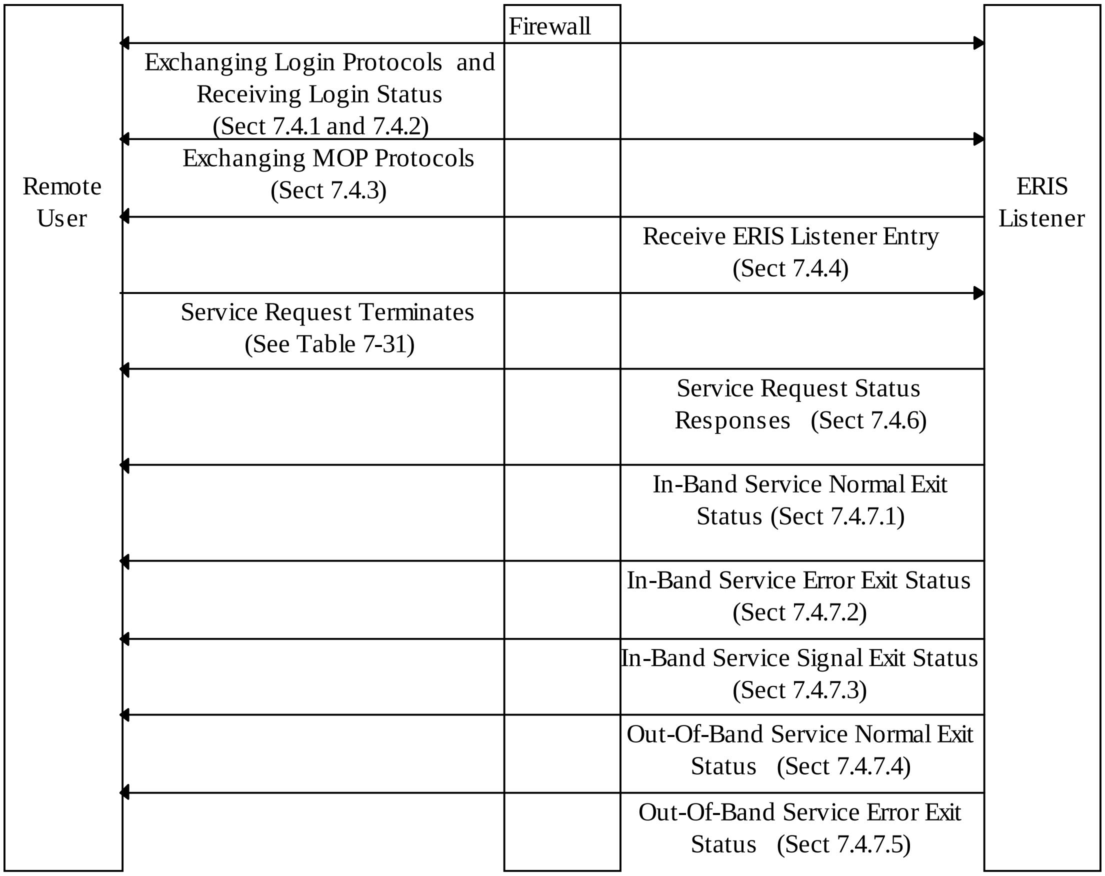

Firewall
Remote User
ERIS Listener
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
Exchanging MOP Protocols (Sect 7.4.3)
Receive ERIS Listener Entry (Sect 7.4.4)
Service Request Terminates (See Table 7-31)
Service Request Status Responses (Sect 7.4.6)
In-Band Service Normal Exit Status (Sect 7.4.7.1)
In-Band Service Error Exit Status (Sect 7.4.7.2)
In-Band Service Signal Exit Status (Sect 7.4.7.3)
Out-Of-Band Service Normal Exit Status (Sect 7.4.7.4)
Out-Of-Band Service Error Exit Status (Sect 7.4.7.5)

Figure 7-7. Receive Service Termination Status

7.4.7.1 IN-BAND SERVICE NORMAL EXIT STATUS

Source: ERIS Listener
Destination: Remote user process

Table 7-42. In-Band Service Normal Exit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	In-Band Service Normal Exit Status Message Code	Char	“0000279025”	Message code for specifying that an In-Band service had a normal exit.
10	1	White space	Char	<space>	Space added to message.
11-19	9	In-Band Service Normal Exit Status Text Part 1	Char	“Service’”	
20 to 20+n-1	1..n	Service Name	Char	Any combination of non-white space characters	

Table 7-42. In-Band Service Normal Exit Status

Byte	Width	Field Name	Type	Values	Description
20+n	1	White space	Char	<space>	Space added to message.
21+n to 21+n+p-1	1..p	Service Arguments	Char	Any combination of words separated by white space	
21+n+p to 33+n+p	13	In-Band Service Normal Exit Status Text Part 2		"' terminated."	
34+n+p to 35+n+p	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279025 Service 'show_common' terminated. \r\n

7.4.7.2 IN-BAND SERVICE ERROR EXIT STATUS

If an In-Band service ended with an error status, the In-Band Service Error Exit Status message will be issued. The Exit code that is returned with this message is dependent upon the In-Band Service that was just exited. For example, if the In-Band Service that exited with an error that occurred from a POIC generated In-Band Service, then the error code would be identified in this document in the section that corresponded to the In-Band Service Request.

Note: Although this capability is offered through ERIS, no known POIC generated software takes advantage of this capability. It is included for future software releases.

Source: ERIS Listener
Destination: Remote user process

Table 7-43. In-Band Service Error Exit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	In-Band Service Error Exit Status Message Code	Char	"0000279011"	Message code for specifying that an In-Band service had an error upon exiting.
10	1	White space	Char	<space>	Space added to message.
11-36	26	In-Band Service Normal Exit Status Text Part 1	Char	"Abnormal completion code '"	
37-39	3	Exit Code		"001" .."255"	

Table 7-43. In-Band Service Error Exit Status

Byte	Width	Field Name	Type	Values	Description
40-62	23	In-Band Service Error Exit Status Text Part 2		“’ received as service ’”	
63-63+ n-1	1..n	Service Name	Char	any combination of non-white space characters	
63+n	1	White space	Char	<space>	Space added to message.
63+n to 63+n +p-1	1..p	Service Arguments	Char	any combination of words separated by white space	
63+n +p to 75+n +p	13	In-Band Service Error Exit Status Text Part 3		“’ terminated."	
76+n +p - to 77+n +p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279011 Abnormal completion code '1' received as service 'show_common ' terminated. \r\n

7.4.7.3 IN-BAND SERVICE SIGNAL EXIT STATUS

If an In-Band service terminated due to receiving an asynchronous notification, the ERIS Listener will issue the In-Band Service Signal Exit Status message. The Signal Exit that is returned with this message is dependent upon the In-Band Service that was just exited. Either a POIC-generated In-Band Service Request or a UNIX system command Service Request can pass through the Signal Exit error received.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-44. In-Band Service Signal Exit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	In-Band Service Signal Exit Status Message Code	Char	“0000279012”	Message code for specifying that an In-Band service had a signal status upon exiting.
10	1	White space	Char	<space>	Space added to message.
11-18	8	In-Band Service Signal Exit Status Text Part 1	Char	“Signal ‘”	
19 - 19+s	1..s	Signal Name		Any combination of alphabetic characters	
20+s-38+s	18	In-Band Service Signal Exit Status Text Part 2		“' caused service '”	
39+s-39+s+n-1	1..n	Service Name	Char	any combination of non-white space characters	
39+n+s	1	White space	Char	<space>	Space added to message.
40+s+n to 40+s+n+p-1	1..p	Service Arguments	Char	Any combination of words separated by white space	
40+n+p+s to 55+n+p+s	15	In-Band Service Signal Exit Status Text Part 3		“' to terminate."	
56+n+p+sto 57+n+p+s	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279012 Signal 'SIGTERM' caused service 'show_common ' to terminate. \r\n

7.4.7.4 OUT-OF-BAND SERVICE NORMAL EXIT STATUS

If an Out-Of-Band service terminated due to receiving an asynchronous notification, the ERIS Listener will issue the Out-Of-Band Service Normal Exit Status message.

Source: ERIS Listener
Destination: Remote user process

Table 7-45. Out-Of-Band Service Exit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Out-Of-Band Service Exit Status Message Code	Char	“0000279026”	Message code for specifying that an Out-Of-Band service is exiting.
10	1	White space	Char	<space>	Space added to message.
11-27	17	Out-Of-Band Service Exit Status Text Part 1	Char	“Service request ‘”	
28-29*	1..2*	Service Request ID		“1” .. n	Maximum value 99
29* - 40	12	Out-Of-Band Service Exit Status Text Part 2		“’ (service '”	
41* to 41+n-1	1..n	Service Name	Char	any combination of non-white space characters	
41*+n	1	White space	Char	<space>	Space added to message.
41*+n to 41+n+p-1	1..p	Service Arguments	Char	any combination of words separated by white space	
41*+n+p to 54+n+p	14	Out-Of-Band Service Exit Status Text Part 3		“') terminated."	
55*+n+p to 56+n+p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

*For cleaner byte specifications, assuming only one character used.
Example Message:
0000279026 Service request ‘13’ (service 'show_common ') terminated. \r\n

7.4.7.5 OUT-OF-BAND SERVICE ERROR EXIT STATUS

If an Out-Of-Band service terminated due to receiving an error code, the ERIS Listener will issue the Out-Of-Band Error Exit message.

Source: ERIS Listener
Destination: Remote user process

Table 7-46. Out-Of-Band Service Error Exit Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Out-Of-Band Error Exit Message Code	Char	“0000279039”	Message code for specifying that an Out-Of-Band service is exiting with an error.
10	1	White space	Char	<space>	Space added to message.
11-36	26	Out-Of-Band Error Exit Status Text Part 1	Char	“Abnormal completion code ‘”	
37-39	3	Error Completion Code		“001”..”999”	Maximum value 999, Values are left justified filled with zeros.
40-70	31	Out-Of-Band Error Exit Status Text Part 2		“’ received as service request '”	
71-72*	1..2*	Service Request ID		“1” .. n	Maximum value 99
*72-83	12	Out-Of-Band Error Exit Status Text Part 3		“’ (service '”	
84* to 84+n-1	1..n	Service Name	Char	any combination of non-white space characters	
84*+n	1	White space	Char	<space>	Space added to message.
85*+n to 85+n+p-1	1..p	Service Arguments	Char	any combination of words separated by white space	
85*+n+p to 98+n+p	14	Out-Of-Band Error Exit Status Text Part 4		“') terminated."	
99*+n+p to 100+n+p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

*For cleaner byte specifications, assuming only one character used.

Example Message:

0000279039 Abnormal completion code '001' received as service request '13' (service 'CDP 111.222.123.234. 1234 ') terminated.\r\n

7.5 STATUS SERVICE REQUEST

Service Type: In-Band

This message allows the remote user process to obtain status on the Out-Of-Band ERIS services running in the current login session. In response, the ERIS Listener will send one of the Service Status Response messages. The Service Status List Response message will be sent if there are Out-Of-Band services running. Otherwise, the No Services Response message will be returned. Figure 7-8 shows the message exchange associated with this status request.

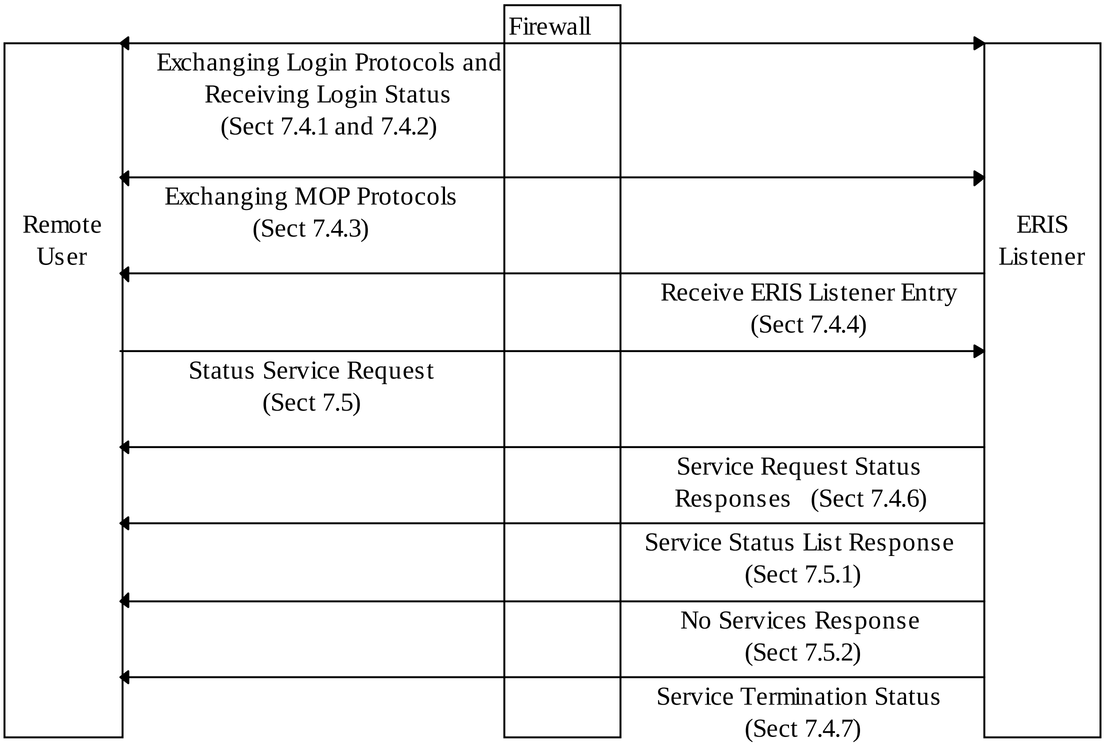

Firewall
Remote User
ERIS Listener
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
Exchanging MOP Protocols (Sect 7.4.3)
Receive ERIS Listener Entry (Sect 7.4.4)
Status Service Request (Sect 7.5)
Service Request Status Responses (Sect 7.4.6)
Service Status List Response (Sect 7.5.1)
No Services Response (Sect 7.5.2)
Service Termination Status (Sect 7.4.7)

Figure 7-8. Exchange Status Request

Source: Remote user process
Destination: ERIS Listener

Table 7-47. Status Service Request

Byte	Width	Field Name	Type	Values	Description
0-5	6	Status Command	Char	“status”	
6	1	Line Terminator	Char	“\n” (ASCII value 10)	New line character

Example Message:
status \n

7.5.1 SERVICE STATUS LIST RESPONSE

The following group of fields will be repeated once for each Out-Of-Band service currently running. The places where the white space is identified by "Tab" indicate that a tab character (ASCII /t decimal 11) is placed at that spot. The Tabs are included to line up the multiple Service Status List Responses so that they appear to be in columns. The List is preceded by a header statement.

Source: ERIS Listener
Destination: Remote user process

This message contains a header line followed by the status list.

Table 7-48. Service Status List Response Header

Byte	Width	Field Name	Type	Values	Description
0-9	10	Informational Message Code	Char	"0000279001"	Message code for providing information only for an interactive user
10	1	White space	Char	<space>	Space added to message.
11-12	2	Service Request ID Label	Char	"ID"	
13-15	3	White space	Char	3 spaces	Space added to message to line up text.
16-25	10	Start Time Label	Char	"Start Time"	
26-47	22	White space	Char	22 spaces	Space added to message to line up text
48-54	7	Service Header	Char	"Service"	
55-56	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:

```
0000279001  ID   Start Time                    Service\r\n
```

Table 7-49. Service Status List Response

Byte	Width	Field Name	Type	Values	Description
0-9	10	Service Status List Response Message Code	Char	"0000279027"	Message code for specifying the status of the list of Out-Of-Band services.
10	1	White space	Char	<space>	Space added to message.
11..13	3	Service Request ID	Char	"001" - "999"	

Table 7-49. Service Status List Response

Byte	Width	Field Name	Type	Values	Description
14	1	White space	Char	<tab>	/t (ASCII Decimal 11)
15 to 17	3	Start Time Weekday	Char	“Mon” “Tue” “Wed” “Thu” “Fri” “Sat” “Sun”	
18	1	White space	Char	<space>	Space added to message.
19 to 21	3	Start Time Month	Char	“Jan” “Feb” “Mar” “Apr” “May” “Jun” “Jul” “Aug” “Sep” “Oct” “Nov” “Dec”	
22	1	White space	Char	<space>	Space added to message.
23 to 24	2	Start Time Month Day	Char	“ 1”..”31” (One-digit numbers are preceded by a space)	
25	1	White space	Char	<space>	Space added to message.
26 to 27	2	Start Time Hour	Char	“00”..”23”	
28	1	Start Time Delimiter	Char	“:”	
29 to 30	2	Start Time Minute	Char	“00”..”59”	
31	1	Start Time Delimiter	Char	“:”	
32 to 33	2	Start Time Second	Char	“00”..”59”	
34	1	White space	Char	<space>	Space added to message.
35 to 37	3	Start Time Zone	Char	“GMT”	
38	1	White space	Char	<space>	Space added to message.

Table 7-49. Service Status List Response

Byte	Width	Field Name	Type	Values	Description
39to 42	4	Start Time Year	Char		Four numeric digits representing the year the service was started
43 to 46	4	White space	Char	4 spaces	Spaces added to message to line up text.
47 to 47+ n-1	1..n	Service Command Name	Char		Any combination of non-white space characters
47+n	1	White space	Char	<space>	Space added to message.
48+n to 48+n +p-1	1..p	Service Command Arguments	Char		Any combination of words separated by white space. The string will be inserted exactly as it was placed on the Service Request input from the Remote User process including spaces.
48+n +p - 49+n +p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:

```
0000279027 12      Wed Feb 10 13:10:00 GMT 1999    CDP  < argument string> \r\n
```

So together they would look like this:

```
0000279001 ID      Start Time                      Service\r\n
0000279027 12      Wed Feb 10 13:10:00 GMT 1999    CDP 111.222.123.234 1234 \r\n
0000279027 10      Tue Feb 9 23:58:32 GMT 1999     CDP 111.222.123.234 1235 r\n
```

7.5.2 NO SERVICES RESPONSE

If the Service Request type is asking for a function to be performed on one or more Out-Of-Band services and there are no Out-Of-Band services currently running, then this response would be given.

Source: ERIS Listener
Destination: Remote user process

Table 7-50. No Services Response

Byte	Width	Field Name	Type	Values	Description
0-9	10	No Services Response Message Code	Char	“0000279028”	Message code for specifying that there are no Out-Of-Band services.
10	1	White space	Char	<space>	Space added to message.

Table 7-50. No Services Response

Byte	Width	Field Name	Type	Values	Description
11-56	46	No Services Response Text	Char	"No Out-Of-Band services are currently running."	
57-58	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000279028 No Out-Of-Band services are currently running. \r\n

7.6 STOP SERVICE REQUEST

Service Type:	In-Band
Source:	Remote user process
Destination:	ERIS Listener

This message allows the remote user process to terminate a service request in the current login session. If the service request ID matches a service running in the current login session, the ERIS Listener will attempt to terminate that service and will thereafter return one of the Service Termination Status messages. If the service request ID is not valid, the Invalid Service Request ID message will be returned. If there are no Out-Of-Band services running, the No Services Response message will be returned.

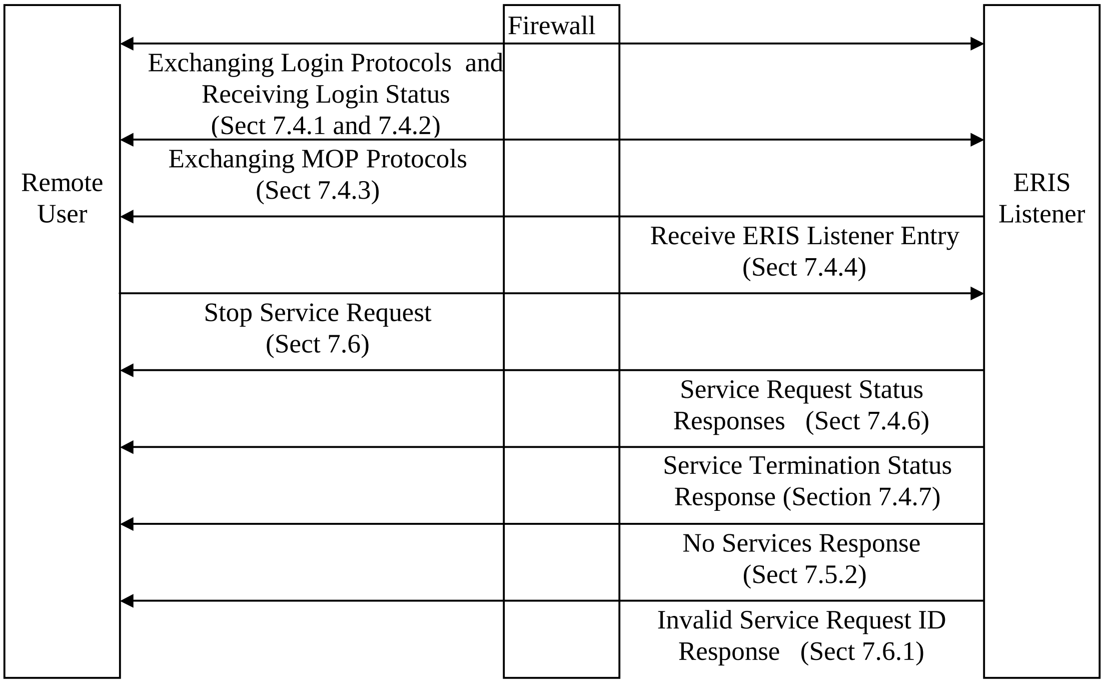

Firewall
Remote User
ERIS Listener
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
Exchanging MOP Protocols (Sect 7.4.3)
Receive ERIS Listener Entry (Sect 7.4.4)
Stop Service Request (Sect 7.6)
Service Request Status Responses (Sect 7.4.6)
Service Termination Status Response (Section 7.4.7)
No Services Response (Sect 7.5.2)
Invalid Service Request ID Response (Sect 7.6.1)

Figure 7-9. Exchange Stop Service Request

Table 7-51. Stop Service Request

Byte	Width	Field Name	Type	Values	Description
0-3	4	Stop Service Command	Char	“stop”	Request to stop service.
4	1	White space	Char	<space>	Parsing searches for next non-white space
5..6	1..2	Service Request ID	Char	“1” .. n	Number of Service Request ID returned when permanent service was started.
7	1	Line Terminator	Char	“\n” (ASCII value 10)	New line character

Example Message:
stop 12\n

7.6.1 INVALID SERVICE REQUEST ID RESPONSE

If the Service Request requires that a Service ID be included in the Service Request and that Service ID is not valid, then this response will be sent to the remote user process.

Source:	ERIS Listener
Destination:	Remote user process

Table 7-52. Invalid Service Request ID Response

Byte	Width	Field Name	Type	Values	Description
0-9	10	Invalid Service Request ID Response Message Code	Char	“0000279029”	Message code for specifying that a service ID was invalid on the request.
10	1	White space	Char	<space>	Space added to message.
11-30	20	Invalid Service Request ID Response Text Part 1	Char	“Service request ID ‘”	
31.. 32	1..2*	Service Request ID		“1” .. n	
32*-72	41	Invalid Service Request ID Response Text Part 2		“’ does not reference any current service.”	
73 - 74	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

*For cleaner byte specifications, assuming only one character used.
Example Message:
0000279029 Service request ID ‘12’ does not reference any current service. \r\n

7.6.2 SERVICE TERMINATION REQUESTED STATUS

If a Stop Service Request has been requested, the ERIS Listener will issue the Service Termination Requested Status message to let the remote user process know that the Stop request has been received and is being processed.

Source: ERIS Listener
Destination: Remote user process

Table 7-53. Service Termination Requested Status

Byte	Width	Field Name	Type	Values	Description
0-9	10	Service Termination Requested Message Code	Char	“0000279041”	Message code for specifying that a Stop Request has been received.
10	1	White space	Char	<space>	Space added to message.
11-55	45	Service Termination Requested Status Text Part 1	Char	“Termination request sent to service request ‘”	
56-57*	1..2*	Service Request ID		“1” .. n	Maximum value 99
*57-68	12	Service Termination Requested Status Text Part 2		“’ (service '”	
69* to 69+n-1	1..n	Service Name	Char	Any combination of non-white space characters	
69*+n	1	White space	Char	<space>	Space added to message.
70*+n to 70+n+p-1	1..p	Service Arguments	Char	Any combination of words separated by white space	
70*+n +p to 73+n+p	3	Service Termination Requested Status Text Part 3		“')."	
74*+n +p to 75+n+p	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

*For cleaner byte specifications, assuming only one character used.

Example Message:

0000279041 Termination request sent to service request '13' (service 'CDP 111.222.123.234 1234 ') terminated.\r\n

7.7 EXIT SERVICE REQUEST

Service Type: In-Band

This message allows the remote user process to terminate the login. The ERIS Listener will then terminate all running Out-Of-Band services and disconnect the login session.

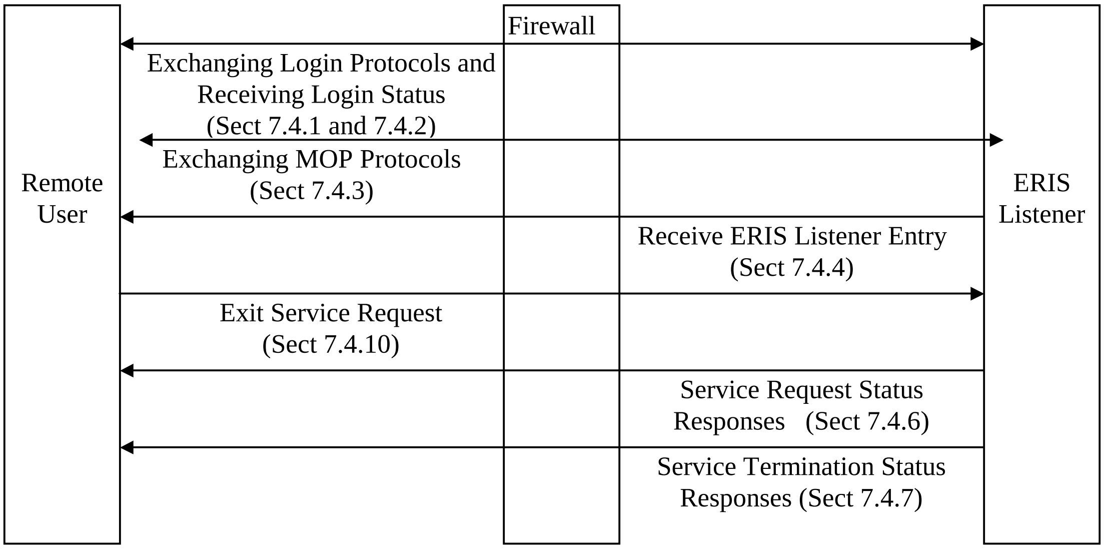

Firewall
Remote User
ERIS Listener
Exchanging Login Protocols and Receiving Login Status (Sect 7.4.1 and 7.4.2)
Exchanging MOP Protocols (Sect 7.4.3)
Receive ERIS Listener Entry (Sect 7.4.4)
Exit Service Request (Sect 7.4.10)
Service Request Status Responses (Sect 7.4.6)
Service Termination Status Responses (Sect 7.4.7)

Figure 7-10. Exchange Exit Service Request

Source: Remote user process
Destination: ERIS Listener

Table 7-54. Exit Command

Byte	Width	Field Name	Type	Values	Description
0-3	4	Exit Command	Char	“exit”	Request to stop all services from running and disconnect the login session.
4	1	Line Terminator	Char	“\n” (ASCII value 10)	New line character

Example Message:
exit\n

7.8 COMMON CONFIGURATION SERVICE REQUEST

Service Type: In-Band

The Common Configuration Service Request will allow the Remote User Process to receive the basic information about the configuration of the ERIS Listener that the Remote user process is logged into at the POIC. The term used for this basic configuration information is called "common configuration" since it is common to all elements of the POIC. Due to the transmission of the data across the ERIS socket (i.e., in-band), all data is represented in an ASCII format for user readability.

Source:	Remote user process
Destination:	ERIS Listener

The "show_common" request is generated by the Remote User process to gather the information associated with the configuration of the ERIS Listener that the Remote user process is currently logged into. Once the request is sent to the ERIS Listener, the information is returned as an In-Band communication.

Table 7-55. Shows Common Request

Byte	Width	Field Name	Type	Values	Description
0-10	11	Common Configuration Request Command	Char	"show_common"	Message code for starting the capability to show the common configuration information.
11	1	Line Terminator	Char	"\n" (ASCII value10)	New line Character

This interface consists of two messages, Show Common Response and Show Common Failure Response, uniquely identified by the first message code received.

7.8.1 SHOW COMMON RESPONSE

The Show Common Response is a combination of different message codes that give the POIC Common Configuration information for the remote user process to use. This information is contained in the following message tables and is received as a response to the "show_common" request.

Source:	Show Common Process
Destination:	Remote user process

One of the prime information components of the common configuration is the MOP associated with the remote user process that the remote user process is logged into. There are many items in the POIC that are MOP dependent. Among them are the databases as identified in the later part of the Common Configuration Response message. The first message line returned from the "show_common" Request is the Project information. The programmatic interface is developed as a generic capability and the ERIS Listener may return MOP configurations to allow the user to log into other projects besides ISS (see section 7.4). If the wrong MOP was chosen, the Project Mission and Operational Support Mode may be different than what was desired by the Remote User process.

Table 7-56. Project Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Project Message Code	Char	"0000244903"	Message code for showing the project information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-17	8	Project label	Char	"Project:"	
18	1	White space	Char	<space>	Space added to message.
19-20 or 19-21	2-4	Project Value		"All" "AXAF" "ISS" "SL" "STS"	
21-22 or 22-23	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244903 Project: ISS\r\n

Within the POIC, another portion of the MOP is the Mission Name. This name is used in the POIC when discussing the current mission or increment that the remote user process is logged into. The Mission Name is alphanumeric and must be four characters in length.

Table 7-57. Mission Name Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Mission Name Message Code	Char	"0000244906"	Message code for showing the mission name information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-23	13	Mission Name Label	Char	"Mission Name:"	
24	1	White space	Char	<space>	Space added to message.
25-28	4	Mission Name Value		any combination of numeric, uppercase alphabetic, and underscores	
29-30	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244906 Mission Name: MSL1\r\n

The Mission ID is the numeric mission identification that is assigned to the Mission internal to the POIC software. There are times when the Mission ID is requested when interfacing with the POIC software.

Table 7-58. Mission ID Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Mission ID Message Code	Char	"0000244909"	Message code for showing the mission identifier information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-21	11	Mission ID Label	Char	"Mission ID:"	
22	1	White space	Char	<space>	Space added to message.
23-25	3	Mission ID Value		"000".."255"	Zero filled on the left.
26-27	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244909 Mission ID: 003\r\n

The final portion of the MOP is the Operational Support Mode (called Operational Mode in this context). The Operational Support Mode identifies whether the POIC is configured for real-time flight mode or set up to handle simulations, test, or other operational support modes.

Table 7-59. Operational Mode Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Operational Mode Message Code	Char	"0000244912"	Message code for showing the operational support mode information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-27	17	Operational Mode Label	Char	"Operational Mode:"	
28	1	White space	Char	<space>	Space added to message.
29-30 or 29-35	2..7	Operational Mode Value		"All" "Flight" "Test" "Sim" "VV" "Dev" "Train" "Offline"	

Table 7-59. Operational Mode Message

Byte	Width	Field Name	Type	Values	Description
31-32 or 36-37	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244912 Operational Mode: Flight\r\n

The following messages are associated with the revisions of the Telemetry Database that are currently being supported at the time of the receipt of the "show_common" request. The Telemetry database information has four different Configuration Management states available to the remote user process. If the particular Configuration Management state is not available at the time of the "show_common" request, then that version number will be identified with dashes (e.g., "----"). The version number is an alphanumeric four-character string that is zero filled (decimal 0) on the left.

The Baseline version of the Telemetry Database is the current version of the Telemetry Database that is being used for the current operational support mode for all User Generated Data Elements.

Table 7-60. Baseline TDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Baseline TDB Version Message Code	Char	"0000244915"	Message code for showing the Baselined TDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-31	21	Baseline TDB Version Label	Char	"Baseline TDB Version:"	
32	1	White space	Char	<space>	Space added to message.
33-36	4	Baseline TDB Version Value		4 alphanumeric characters or "----" (i.e., undefined)	
37-38	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244915 Baseline TDB Version: 0023\r\n

The Pre-released version of the Telemetry Database becomes available prior to a new Telemetry Database version becoming baselined. It is probable that there is not a pre-released version of the Telemetry Database available at the time of the "show_common" request. The pre-released version is to be used to prepare the remote user process for the subsequent baselining of the pre-released version.

Table 7-61. Pre-release TDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Pre-release TDB Version Message Code	Char	"0000244918"	Message code for showing the Pre-released TDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-33	23	Pre-release TDB Version Label	Char	"Pre-release TDB Version:"	
34	1	White space	Char	<space>	Space added to message.
35-38	4	Pre-release TDB Value		4 alphanumeric characters or "----" (i.e., undefined)	
39-40	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244918 Pre-release TDB Version: 0024\r\n

The Archived Database is a previously "Baselined" version of the Telemetry Database that is mounted for use for playbacks of telemetry data. The POIC only has one archived database loaded at a time. It is possible for there not to be an Archived Telemetry Database loaded at the time of the "show_common" request.

Table 7-62. Archive TDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Archive TDB Version Message Code	Char	"0000244921"	Message code for showing the Archived TDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-30	20	Archive TDB Version Label	Char	"Archive TDB Version:"	
31	1	White space	Char	<space>	Space added to message.
32-35	4	Archive TDB Version Value		4 alphanumeric characters or "----" (i.e., undefined)	
36-37	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244921 Archive TDB Version: 0021\r\n

The Delivered Telemetry Database is the version of the database that has been delivered to the POIC to be verified. The remote user process should use this version of the database at their own risk. This version of the database is not intended for anything but preliminary working with a new database. It is probable that the Delivered TDB Version will not be available.

Table 7-63. Delivered TDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Delivered TDB Version Message Code	Char	"0000244924"	Message code for showing the Delivered TDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-32	22	Delivered TDB Version Label	Char	"Delivered TDB Version:"	
33	1	White space	Char	<space>	Space added to message.
34-37	4	Delivered TDB Version Value		4 alphanumeric characters or "---- " (i.e., undefined)	
38-39	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244924 Delivered TDB Version: 0031\r\n

The following messages are associated with the revisions of the Command Database that are currently being supported at the receipt of the "show_common" request. The Command database information has three different Configuration Management states available to the remote user process. If the particular Configuration Management state is not available at the time of the "show_common" request, then that version number will be identified with dashes (e.g., "----"). The version number is an alphanumeric four character string that is zero (decimal 0) filled on the left.

The baseline version of the Command Database is the version of the database being used to create commands to send to the ISS.

Table 7-64. Baseline CDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Baseline CDB Version Message Code	Char	"0000244927"	Message code for showing the Baselined CDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-31	21	Baseline CDB Version Label	Char	"Baseline CDB Version:"	
32	1	White space	Char	<space>	Space added to message.

Table 7-64. Baseline CDB Version Message

Byte	Width	Field Name	Type	Values	Description
33-36	4	Baseline CDB Version Value		4 alphanumeric characters or "----" (i.e., undefined)	
37-38	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244927 Baseline CDB Version: 0013\r\n

The pre-released version of the Command Database becomes available prior to a new Command Database version becoming baselined. It is probable that there is not a pre-released version of the Command Database available at the time of the "show_common" request. The pre-released version is to be used to prepare the remote user process for the subsequent baselining of the pre-released version.

Table 7-65. Pre-release CDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Pre-release CDB Version Message Code	Char	"0000244930"	Message code for showing the Pre-released CDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-33	23	Pre-release CDB Version Label	Char	"Pre-release CDB Version:"	
34	1	White space	Char	<space>	Space added to message.
35-38	4	Pre-release CDB Version Value		4 alphanumeric characters or "----" (i.e., undefined)	
39-40	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244930 Pre-release CDB Version: 0014\r\n

The Delivered Command Database is the version of the Command Database that has been delivered to the POIC to be verified. The remote user should use this version of the database at their own risk. This version of the database is not intended for anything but preliminary working with a new database. It is probable that the Delivered CDB Version will not be available.

Table 7-66. Delivered CDB Version Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Delivered CDB Version Message Code	Char	“0000244933”	Message code for showing the Delivered CDB Version information associated with the Common Configuration
10	1	White space	Char	<space>	Space added to message.
11-32	22	Delivered CDB Version Label	Char	“Delivered CDB Version:”	
33	1	White space	Char	<space>	Space added to message.
34-37	4	Delivered CDB Version Value		4 alphanumeric characters or “----“ (i.e., undefined)	
38-39	2	Line Terminator	Char	“\r\n” (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
```
0000244933 Delivered CDB Version: 0018\r\n
```

Since all of the above messages are returned from the single “show_common” request, the actual example of the outcome string of messages would look like this:

```
0000244903 Project: ISS\r\n
0000244906 Mission Name:MSL1\r\n
0000244909 Mission ID:003\r\n
0000244912 Operational Mode: Flight\r\n
0000244915 Baseline TDB Version: 0023\r\n
0000244918 Pre-release TDB Version: 0024\r\n
0000244921 Archive TDB Version: 0021\r\n
0000244924 Delivered TDB Version: 0031\r\n
0000244927 Baseline CDB Version: 0013\r\n
0000244930 Pre-release CDB Version: 0014\r\n
0000244933 Delivered CDB Version: 0018\r\n
```

7.8.2 SHOW COMMON FAILURE RESPONSE

If the Show Common Failure Response message is received, this indicates an I/O subsystem error. Please contact the HOSC Help Desk (256-544-5066) to report the error.

Table 7-67. Show Common Failure Response Message

Byte	Width	Field Name	Type	Values	Description
0-9	10	Show Common Request Failure Message Code	Char	"0000244900"	Message code for showing failure in getting the common configuration information.
10	1	White space	Char	<space>	Space added to message.
11-28	18	Common Request Failure Message Label	Char	"show_common FAILED"	
29-30	2	Line Terminator	Char	"\r\n" (ASCII value 13 followed by ASCII value 10)	Carriage Return followed by New line character

Example Message:
0000244900 show_common FAILED\r\n

7.9 CUSTOM DATA PACKET SERVICE REQUEST

Service Type: Out-of-Band

This message allows the remote user process to start the CDP process. It does not respond with any unique messages.

Source: Remote user process
Destination: ERIS Listener

Table 7-68. Start CDP Service Request

Byte	Width	Field Name	Type	Values	Description
0-2	3	Custom Data Packet Request Command	Char	"CDP"	
3	1	White space	Char	<space> or <tab>	Space added to message.
4-3+n	1..n	Host Address	Char	any combination of characters	Include dotted decimal IP address for remote user GSE. Note: Value must have been pre-approved in the POIC account request form.
4+n	1	White space	Char	<space> or <tab>	Space added to message.
5+p-4+n +p	1..p	Port Number	Char	interpreted as an integer	Port for the Custom Data Packet Socket #1 for control information to connect back to remote user. Note: Value must have been pre-approved in the POIC account request form.

Table 7-68. Start CDP Service Request

Byte	Width	Field Name	Type	Values	Description
5+n +p	1	Line Terminator	Char	"\n" (ASCII value 10)	New line character

Example Message:
CDP 111.222.123.234 1234\n

To stop a CDP, use the Stop Service Request. See Section 7.6, Stop Service Request, for how to stop the service.

7.10 REMOTE COMMAND SERVICE REQUEST

Service Type: Out-of-Band

This message allows the remote user process to start the command process. It does not respond with any unique messages.

Source: Remote user process
Destination: ERIS Listener

Table 7-69. Start Remote Command Service Request

Byte	Width	Field Name	Type	Values	Description
0-6	7	Remote Command Request	Char	"command"	
7	1	White space	Char	<space> or <tab>	Space added to message.
8-7+n	1..n	Host Address	Char	Any combination of characters	Include dotted decimal IP address for remote user GSE. Value must have been pre-approved in the POIC account request form.
8+n	1	White space	Char	<space> or <tab>	Space added to message.
9+n-8+n +p	1..p	Port Number	Char	interpreted as an integer	Port for the remote commanding socket for control information to connect back to remote user. Value must have been pre-approved in the POIC account request form.
9+n +p	1	Line Terminator	Char	"\n" (ASCII value 10)	New line character

Example Message:
Command 111.222.123.234 1234\n

To stop a remote command, use the Stop Service Request. See Section 7.6, Stop Service Request, for how to stop the service.

8.0 TELEMETRY PROTOCOLS

This section describes the detailed telemetry protocols associated with telemetry that is distributed by the POIC. Users requiring delivery of any of these telemetry streams need to specify their requirements in the PDL. The protocols described in this section include:

a. The overall Enhanced HOSC System (EHS) protocol

b. Consultative Committee for Space Data System (CCSDS) packet headers

c. Payload Data Services System (PDSS) Payload CCSDS packet

d. PDSS Payload Bitstream Protocol Data Unit (BPDU)

e. PDSS User Data Summary Message

f. PDSS Retrieval Processing Summary Message

g. Ground Support Equipment GSE packet protocol, which is used by the Ground Ancillary GSE Packet(s)

h. CDP

8.1 ENHANCED HOSC SYSTEM (EHS) PROTOCOL

The EHS protocol is designed to facilitate the identification and distribution of data throughout the POIC and to remote locations. The EHS protocol utilizes the UDP (connectionless) transport mechanism for all of the telemetry protocols, with the exception of Custom Data Packets that utilize TCP. Figure 8-1, Protocol Stack, defines the protocol stack in a bottom to top orientation with the size restrictions displayed on the right. The first row shown is the IP and the second layer is a combination of UDP and TCP. To determine which protocols use either UDP or TCP see the protocols defined on top of these layers. This protocol stack further identifies whether the protocol uses either the CCSDS packet format or the BPDU.

<table>
<tr><td>Custom Data
Packet
(section 8.7)</td><td>PDSS Payload CCSDS Packet (section 8.2)
PDSS BPDU (section 8.3)
PDSS UDSM (section 8.4)
PDSS RPSM (section 8.5)
GSE Packet (section 8.6)</td></tr>
<tr><td colspan="2"><b>EHS Protocol (Section 8.1)</b></td></tr>
<tr><td><b>TCP Protocol
(section 3.3.2.3.1)</b></td><td><b>UDP Protocol (Section 3.3.2.3.2)</b></td></tr>
<tr><td colspan="2"><b>IP Protocol (section 3.3.2.2.1)</b>
LLC Protocol
MAC Protocol
PHY Protocol (section 3.3.2.1)</td></tr>
</table>

Figure 8-1. Protocol Stack

The generic message structure of the EHS protocol is divided into two areas: the Primary EHS Protocol Header area and the EHS Protocol Data area. These are shown in Figure 8-2 in a left to right orientation, which is the structure used on the remainder of relevant figures in this section. The type of EHS Protocol Data contained in the structure is determined by the value of the Secondary Protocol Header Type field in the Primary EHS Protocol Header (see Table 8-1). The EHS Protocol Data will contain a Secondary EHS Protocol Header followed by either a CCSDS Packet or a BPDU. The maximum length of the packets/BPDUs is limited by the internal POIC/PDSS design. The largest packet size supported by POIC is specified in Figure 8-2; however, each section that discusses Secondary EHS Protocol Header types will also further define the limits of that specified type.

<table>
<tr><td>Primary EHS
Protocol Header
(16 bytes)</td><td>EHS Protocol Data
(Maximum 61422 bytes)</td></tr>
<tr><td colspan="2">EHS Protocol - Maximum 61438 bytes</td></tr>
</table>

Figure 8-2. EHS Protocol Structure

8.1.1 PRIMARY EHS PROTOCOL HEADER

The Primary EHS Protocol Header has a fixed length of 16 bytes (128 bits). The Primary EHS Protocol Header format is defined in Table 8-1. Note that the numeric order for the labeling of all bits within bytes/words is from Most Significant Bit (MSB) to Least Significant Bit (LSB), meaning the LSB is bit 0.

Table 8-1. Primary EHS Protocol Header Format

Byte	Bit	Field Name	Type	Values	Description
0	7-4	EHS Protocol Version ID	Integer	0-15 (Decimal), Set to 2.	Identifies the version of the EHS protocol.
	3-0	Project Identifier	Binary	0000 - All projects 0001 - STS[1] 0010 - SL[1] 0011 - ISS 0100 - AXAF[1]	Identifies the originating project. ISS will be the only project identified for the remote user. Set to 0011.
1	7-4	Operational Support Mode	Binary	0000 - All support modes 0001 - Flight 0010 - Ground Test 0011 - Simulation 0100 - Verification /Validation 0101 - Development 0110 - Training 0111 - Off-line	Identifies the type of operation for which this data was generated.
	3-0	Data Mode	Binary	0000 - unused 0001 - Real-Time 0010 - Dump 1 0011 - Dump 2 0100 - Dump 3 0101 thru 1111 Playbacks 1 thru 11	Identifies the mode of the data for telemetry (unused for non-telemetry).

Table 8-1. Primary EHS Protocol Header Format

Byte	Bit	Field Name	Type	Values	Description
2	7-0	Mission/Increment	Integer	0 = all missions, 1 thru 255 (decimal) = the Mission/Increment	ISS increment number.
3	7-0	Secondary Protocol Header Type	Binary	00000000 - All Protocols[1] 00000001 - TDM Telemetry[1] 00000010 - 4800-bit NASCOM Block[1] 00000011 - Pseudo Telemetry[1] 00000100 - TDS Packet (Time) [1] 00000101 - EHS Test Data [1] 00000110 - GSE Packet Data 00000111 - Custom Data Packet 00001000 - HRDS DQ [1] 00001001 - CSS [1] 00001010 - PDSS AOS/LOS Indicator Packet[1] 00001011 - PDSS Payload CCSDS Packet 00001100 - PDSS Core CCSDS Packet[1] 00001101 - PDSS Payload BPDU 00001110 - PDSS UDSM 00001111 - PDSS RPSM	Identifies the content of the EHS Protocol Data area.
4	7-0	Ground Receipt Time (GRT) year	Integer	0-255 (Decimal)	Time of packet origination. For PDSS this will be the time from the IRIG-B time signal received. Time EHS is the number of years since 1900
5-6	15-0	GRT day of year	Integer	1-366 (Decimal), bits 15-9 are always 0	
7	7-0	GRT hours	Integer	0-23 (Decimal), bits 7-5 are always 0	
8	7-0	GRT minutes	Integer	0-59 (Decimal), bits 7-6 are always 0	
9	7-0	GRT seconds	Integer	0-61 (Decimal), bits 7-6 are always 0[2]	
10	7-4	GRT tenths of seconds	Integer	0-9 (Decimal)	
	3	GRT status New Data		0=Old Data 1=New Data	'New Data/Old Data' to indicate when a time has been updated or is stale
	2	Unused		Set to 0.	Reserved for future use.
	1	HOLD Condition	Binary	0=No-HOLD 1=HOLD	'No-Hold/Hold' to indicate a hold condition such as in a launch hold and the value is not updating.

Table 8-1. Primary EHS Protocol Header Format

Byte	Bit	Field Name	Type	Values	Description
	0	Time Value Sign	Binary	0=Positive 1=Negative	'Sign' to indicate a normal time (positive) or a count down time (negative)
11	7-6	Miscellaneous Status Backup Source Id	Binary	0=Not Source Specific 1=Source A 2=Source B	Contains status fields used by multiple EHS protocols. Indicates for streams that are redundantly transmitted if it is from one source or the other (Normally not used for PDSS telemetry data distribution.) Set to 0.
	5-0	unused	Binary	Set to 0	Reserved for future use.
12-13	15-0	unused	Integer	Set to 0	Two octets reserved for future expansion.
14-15	15-0	HOSC Packet Size	Integer	16-61438 (Decimal)	The length of the entire HOSC packet (EHS protocol) in bytes including both the Primary EHS Protocol Header and EHS Protocol Data.

[1]Not applicable to ISS remote users. [2]Allows for 2 leap seconds.

Various fields in the Primary EHS Protocol Header deserve more discussion concerning their definition and use. The main purpose for these discussions is to give the remote user insight into why the different fields are included in the header. The first field of interest is the EHS Protocol Version ID. This field is included in the Primary EHS Protocol Header to allow for future expansion of the Primary EHS Protocol Header. If a change occurs to the Primary EHS Protocol Header that would not be compatible with older versions of the Primary EHS Protocol Header, then this field value would be changed. When dealing with the Primary EHS Protocol Header it would be wise to check this value with your code that receives this Protocol Header and confirm that the header format has not changed.

The next element in the Primary EHS Protocol Header is the Project field. The Primary EHS Protocol Header is used for multiple projects within the POIC. Even though each individual project should only receive the data that are intended for them, it would not hurt to confirm that the value on the Project field matches your project. The project field value is set to 0000 (All Projects) when the data message that is received applies to all projects.

The next three fields in the Primary EHS Protocol header are the Operational Support Mode, the Data Mode, and the Mission or Increment field. These three fields when used together provide the configuration of the telemetry data that are being sent to the remote user. The Operational Support Mode will let the remote user know if the data being received are from flight (i.e., on-orbit) data or whether these data are from a simulation or other activity from the POIC.

The Data Mode should be checked by the remote user's software interface to confirm whether the data being received from the POIC are real-time, dump, or a playback. Table 8-2 indicates the type of packet, the contents of the packet, the assigned data mode, and whether the data mode is applicable to the remote user. The dump mode indicates that the packet has been stored in the Communication Outage Recorder (COR) onboard the ISS. Each type of dump mode is assigned to a particular type of packet. Dump1 indicates that the packet is an S-band Zone of Exclusion (ZOE) Dump packet that has been processed by the ZOE Dump process within the POIC. Dump2 indicates that the packet is a Ku-Band packet that has been temporarily stored in the COR onboard the ISS. All Ku-Band packets generated by payloads (e.g., Payload science packets) and the Payload MDM, such as the payload health and status and flight ancillary packets that have been stored in the COR, are assigned to this data mode. Dump3 indicates that the packet is an S-band ZOE packet that is transmitted via the Ku-band system and has been temporarily stored in the Communication Outage Recorder onboard the ISS.

The assigned data modes for playback packets are a combination of statically and dynamically assigned data modes. The first six playback channels are statically assigned. Playback channel number 1 is assigned to stored Orbiter Downlink telemetry from the Orbiter Interface Unit (OIU) and the Mini Pressurized Logistics Module (MPLM). Playback channel number 2 is assigned to S-band ZOE Dump packets that have been stored temporarily in a Line Outage Recorder at White Sands and sent to the POIC for ZOE Dump processing on the ground. Playback channel number 3 is assigned to Ku-Band packets that have been temporarily stored on the PDSS Line Outage Recorder. Playback channel number 4 is assigned to S-Band packets that have been stored temporarily in a Line Outage Recorder at White Sands and sent to the POIC. Playback channel number 5 is assigned to Ku-band packets that have been stored on the COR and then temporarily stored on the PDSS LOR. Finally, playback channel number 6 is assigned to S-Band packets that are transmitted via the Ku-Band system and temporarily stored in the PDSS LOR.

The dynamically assigned playback channels, numbered 7 thru 11, are generated upon request by the remote user for playback of packets stored either in short-term storage within EHS or long-term storage within PDSS. The remote user will receive notification of the assigned playback channel number via the following applications:

- Message handler application if using X-windows, or
- EHS playback status utility if using X-windows, or
- Playback Status application on the Web (available upon delivery of Build 5.0).

Table 8-2. Data Mode Mapping

Type of Packet	Contents of Packets	Data Mode from PDSS	Applicable to Remote User
Ku-Band Real-time	Payload Science and Engineering Data Payload Health and Status, Broadcast Ancillary Data, Payload Ancillary Data	Real-Time	Yes
S-Band Real-time	Essential, Housekeeping 1, Housekeeping 2	Real-time	No
Orbiter Downlink Real-time	OIU, MPLM	Real-time	Some payloads
S-Band ZOE Real-time	Essential, Housekeeping 1, Housekeeping 2 ZOE	Real-time	No
S-Band ZOE Dump	Essential, Housekeeping 1, Housekeeping 2 ZOE	Dump 1	No
Ku-Band Stored in COR	Payload Science and Engineering Data,	Dump 2	Yes

Table 8-2. Data Mode Mapping

Type of Packet	Contents of Packets	Data Mode from PDSS	Applicable to Remote User
	Payload Health and Status, Broadcast Ancillary Data, Payload Ancillary Data		
S-band ZOE downlinked via Ku-band	Essential, Housekeeping 1, Housekeeping 2 ZOE	Dump 3	No
Orbiter Downlink from JSC Dump Process	OIU, MPLM	Playback Channel 1	No
S-Band ZOE Dump from White Sands	Essential, Housekeeping 1, Housekeeping 2 ZOE	Playback Channel 2	No
Ku-Band Playback from PDSS LOR	Payload Science and Engineering Data, Payload Health and Status, Broadcast Ancillary Data, Payload Ancillary Data	Playback Channel 3	Yes
S-Band Playback from White Sands	Essential, Housekeeping 1, Housekeeping 2 ZOE	Playback Channel 4	No
Ku-band Dump Packets from PDSS LOR	Payload Science and Engineering Data, Payload Health and Status, Broadcast Ancillary Data, Payload Ancillary Data	Playback Channel 5	Yes
S-Band packets downlinked via Ku-band and transmitted from PDSS LOR	Essential, Housekeeping 1, Housekeeping 2 ZOE	Playback Channel 6	No
Short Term and Long Term Playbacks from EHS and PDSS	Any packets that the user is allowed to access.	Playback Channels 7-11	Yes

The Mission or Increment field can be arbitrarily assigned and does not have to be tied to a specific increment.

Remote users should understand the importance of the Secondary Protocol Header Type field. The structure of the Secondary Headers of the EHS Protocol is not required to follow any format restrictions. Any similarities between protocols are purely coincidental and should not be relied upon when developing interfaces to the POIC. Another point to consider is that the Application Process Identifier (APID) within the CCSDS Primary Packet Header (which ISS uses to uniquely identify elements) is not guaranteed to be unique across POIC secondary header types. Secondary Protocol Header Types that are created at the POIC are assigned a POIC-generated APID. There is no guarantee that this APID will be unique from ISS generated APIDs. This is because the Secondary Protocol Header Type combined with the APID uniquely identifies the elements. Thus, when interfacing with data received from the POIC, it is important for the remote user to check the Secondary Protocol Header Type field to determine the type of data being received from the POIC before continuing to process that data internal to their software.

The Ground Receipt Time fields are filled in by the receiving system to indicate the time the data message was received at the POIC. This allows the data message to be time ordered by the POIC receipt time when it is retrieved from the POIC long-term and short-term storage. This timeframe is not the same time that is in the CCSDS secondary header for the onboard time stamp. Data that are generated from the POIC will fill in the GRT value as follows:

a. PDSS payload CCSDS Packets and PDSS Payload BPDUs GRT for real-time data will be the time that the data were received at the POIC.

b. PDSS Payload CCSDS Packets and PDSS Payload BPDUs GRT for COR/dump data will be the time that the data were received at the POIC, not the COR recorded time.

c. UDSM GRT for real-time data will be the time that the UDSM data packet was generated at the POIC.

d. UDSM GRT for COR/dump data will be the time that the UDSM data packet was generated at the POIC.

e. UDSMs generated for test, simulation, or playback activities will use the POIC ground simulated time that corresponds with that activity.

f. Ground Ancillary GSE Packets and User-generated GSE Packets GRT for real-time, COR, dump, or playback data will be dependent upon the value specified by the Payload Operations Integration Facility (POIF) or user respectively when starting the distribution of the GSE Packet.

  - POIC Real-time ground time that the GSE Packet was created.
  - Embedded Measurement Stimulation and Identifier (MSID) time of the user-specified MSID.

g. Ground Ancillary GSE Packets and User-generated GSE Packets GRT generated for test or simulation activities will use either

  - POIC ground simulated time that corresponds with the creation time of that GSE Packet.
  - Embedded MSID time of the user-specified MSID.

h. Custom Data Packets GRT will be the POIC ground creation time of the Custom Data Packet.

The status fields are defined sufficiently by their description definitions. The final field is the HOSC Packet Size. This value is set even if the “packet” is a BPDU that is not a packet. The value includes the header put on by the POIC and the original size of the CCSDS Packet or BPDU.

8.1.2 EHS PROTOCOL DATA

The EHS Protocol Data area will consist of a Secondary EHS Protocol Header followed by the data in a CCSDS packet or a BPDU as shown in Figure 8-3. There are several different types of Secondary EHS Protocol Headers for the following data types that support ISS that are available to remote users:

Stream	Protocol Types	Section
Payload Data Service System (PDSS) Payload CCSDS Packet Data	11 (dec)	8.2
PDSS payload Bitstream Protocol Data Unit (BPDU)	13 (dec)	8.3
PDSS User Data Summary Message (UDSM)	14 (dec)	8.4
PDSS Retrieval Processing Summary Message (RPSM)	15 (dec)	8.5
Ground Support Equipment (GSE) Data Packet	6 (dec)	8.6
Custom Data Packet (CDP)	7 (dec)	8-7

The CCSDS packet consists of a CCSDS primary header, followed by an optional CCSDS Secondary Header, followed by the actual data to complete the CCSDS packet. The BPDU is not further defined.

The format and fields within the Secondary EHS Protocol Header are not guaranteed to be consistent between all the different types of Secondary EHS Protocol types. The APIDs that are unique within a Secondary EHS Protocol type are not unique across other Secondary EHS Protocol types. The value of the Secondary EHS Protocol Header in the Primary EHS Protocol Header must be checked to determine how to access the fields within the Secondary EHS Protocol Header and also determine the type of packet.

Within the Telemetry Format Standard (MSFC-STD-1274), Volume 2, are limitations imposed on the definition of the Primary Header structure. The Protocol data definitions within this document comply with those limitations on the fields within the CCSDS Primary Header. The sum total of the Primary EHS Protocol Header and the Secondary EHS Protocol Header will not exceed 512 bytes.

<table>
<tr><td></td><td></td><td colspan="3">BPDU or</td></tr>
<tr><td colspan="2">Maximum 512 Bytes</td><td colspan="3">CCSDS Packet (Maximum 60926 Bytes)</td></tr>
<tr><td>Primary EHS Protocol Header (16 Bytes)</td><td>Secondary EHS Protocol Header</td><td>CCSDS Primary Header</td><td>CCSDS Secondary Header (optional)</td><td>CCSDS Data Zone</td></tr>
<tr><td></td><td colspan="4">EHS Protocol Data (Maximum 61422 bytes)</td></tr>
</table>

Figure 8-3. EHS Protocol Data Structure with CCSDS Packet

8.2 PAYLOAD DATA SERVICE SYSTEM (PDSS) PAYLOAD CCSDS PACKET

The PDSS Payload CCSDS packet contains payload data downlinked from the ISS. The remote user can receive PDSS Payload CCSDS packets through the PDSS distribution of real-time data. In order to access this real-time distribution of data, the remote user must specify the requirement to access this telemetry through the PDL requirements process as defined in section 1.3. The telemetry distributed by PDSS is in packet format using UDP (section 3.3.2.3.2).

When the Secondary Protocol Header Type field in the Primary EHS Protocol Header is set to decimal value 11, the packet contains PDSS payload CCSDS telemetry. Figure 8-5 shows the EHS Protocol Data structure for PDSS payload CCSDS telemetry. The EHS protocol data area contains a unique secondary protocol header definition followed by a CCSDS packet definition. The maximum number of bytes allowed in the PDSS Payload CCSDS Telemetry according to Figure 8-5, EHS Protocol Data Structure for PDSS Payload CCSDS Telemetry, is based solely upon the maximum size of the packet minus the size of the headers as defined in this document. The actual limitation on the PDSS Payload CCSDS Telemetry packets is limited by onboard ISS to be no more than 4096 bytes.

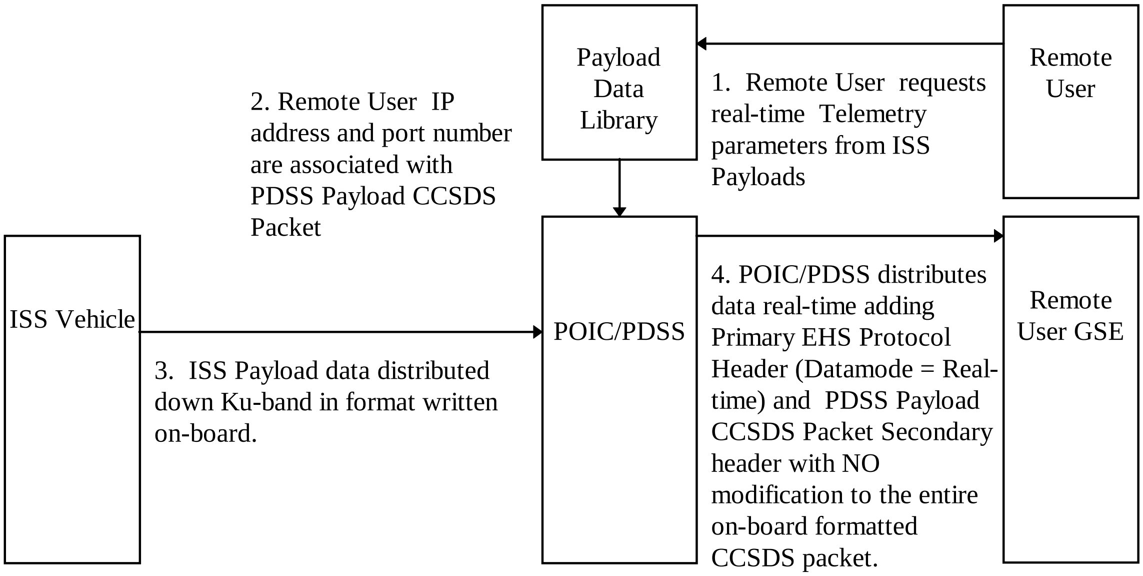

Payload Data Library
Remote User
1. Remote User requests real-time Telemetry parameters from ISS Payloads
2. Remote User IP address and port number are associated with PDSS Payload CCSDS Packet
ISS Vehicle
POIC/PDSS
Remote User GSE
3. ISS Payload data distributed down Ku-band in format written on-board.
4. POIC/PDSS distributes data real-time adding Primary EHS Protocol Header (Datamode = Real-time) and PDSS Payload CCSDS Packet Secondary header with NO modification to the entire on-board formatted CCSDS packet.

Figure 8-4. Receiving PDSS Payload CCSDS Packets

<table>
<tr><td></td><td></td><td colspan="3">CCSDS Packet (Maximum 61,410 Bytes)
(ISS On-board limit to 4096 bytes)</td></tr>
<tr><td>Primary EHS Protocol Header (16 bytes)</td><td>Secondary EHS Protocol Header for PDSS (12 bytes)</td><td>CCSDS Primary Header (6 bytes)</td><td>CCSDS Secondary Header (10bytes)</td><td>CCSDS Data (ISS Payload Data)</td></tr>
<tr><td></td><td colspan="4">EHS Protocol Data</td></tr>
</table>

Figure 8-5. EHS Protocol Data Structure for PDSS Payload CCSDS Telemetry

8.2.1 PRIMARY EHS PROTOCOL HEADER FOR PDSS PAYLOAD CCSDS PACKET

See section 8.1.1 for definition of the Primary EHS Protocol Header.

8.2.2 SECONDARY EHS PROTOCOL HEADER FOR PDSS PAYLOAD CCSDS PACKET

The Secondary EHS Protocol header for PDSS telemetry shall be 12 bytes (96 bits) in length. Table 8-3 defines the Secondary Protocol Header format for PDSS payload CCSDS telemetry.

Table 8-3. Secondary EHS Protocol Header Format for PDSS Payload CCSDS Telemetry

Byte	Bit	Field Name	Type	Values	Description
0	7-6	Version Number	Binary	Set to 00.	This is the PDSS secondary header version number. This number is independent of the EHS protocol version number in the primary header of the EHS protocol.
	5-3	Data Status Unused	Binary	Set to all zeros.	Unused
	2	VCDU Sequence Error	Binary	0=No error 1=Error occurred	Indicates if a VCDU Sequence Error was detected. May indicate why a packet sequence error occurred.
	1	Packet Sequence Error	Binary	0=No error 1=Error occurred	Indicates if a Packet Sequence Error was detected. Used to indicate that packet data was lost
	0	Unused	Binary	Set to Zero.	Unused.
1-3	23-0	Virtual Channel Data Unit (VCDU) Sequence Number	Integer	Value set from VCDU Sequence Number from the VCDU Header	This is the VCDU sequence number taken from the VCDU header. The value is not changed by PDSS.
4	7	Data Stream Identifier (DSID) CCSDS vs. BPDU Identifier	Binary	Set to zero because it is a CCSDS packet.	Fields from Bytes 4 -7 make up the Data Stream Identifier (DSID). This is a unique identifier for the data stream. This bit determines if the type of data contained therein is CCSDS packet data (0) or BPDU (1).
	6-0	Unused	Binary	Set to all zeros.	Reserved for future use.
5	7-0	Unused	Binary	Set to all zeros.	Reserved for future use
6	7-5	Unused	Binary	Set to all zeros.	Reserved for future use.
	4	DSID GSE Packet Identifier	Binary	Set to zero.	Set to 0 to specify this is not a POIC created GSE packet.
	3	DSID Payload vs. Core Identifier	Binary	0=Core (Systems) data 1=Payload data	Corresponds to the type bit and is set to 1 for payload data and set to 0 for core (systems) data.
	2-0	DSID APID MSB	Integer	0-2047 (Decimal)	The eleven bits correspond to the user's APID.
7	7-0	DSID APID LSB			
8-9	15-0	Routing ID	Integer	PDSS internal routing key	This is a routing key used internal to PDSS and should not be used by the remote user This value will be of a dynamic nature.
10-11	15-0	PDSS Reserved	Binary	Set to 0111 1011 1101 1110 (or 7BDE hex.)	This key is used internally to PDSS and should not be used by the remote user. This is a fixed 16-bit pattern.

8.2.3 CCSDS PACKET FOR PDSS PAYLOAD CCSDS PACKET

The CCSDS packet is composed of a standard CCSDS Primary Header, a CCSDS Secondary Header, and ISS payload data. The format of the headers in this CCSDS packet is compliant with MSFC-STD-1274, Volume 2. If the data in the PDSS Payload CCSDS Packet are not compliant with MSFC-STD-1274, Volume 2, it will not be processed by the POIC. The data in the packet would just be throughput to the remote user.

The PDSS Payload CCSDS Packet is not modified by the POIC. The packet is simply reassembled and distributed. No data are added or modified from the original onboard content within the PDSS Payload CCSDS Packet. Any modifications are made to the EHS Primary Protocol Header (e.g., Ground Receipt Time, Mission, Operational Support Mode, Data mode are added).

8.2.3.1 CCSDS PACKET HEADERS FOR PDSS PAYLOAD CCSDS PACKET

The format for the CCSDS Primary Header is consistent with Software Interface Control Document, Part 1, United States On-Orbit Segment (USOS) to International Ground System Segment Ku-Band Telemetry Formats (SSP 41158), Table 4.1.1-1, CCSDS Primary Header Field Definitions. These data are not modified by the POIC system. Note that SSP 41158 sometimes uses the term “Octets” (8 bits), while the following tables use the term Bytes (8 bits) to be consistent with the rest of this document.

Table 8-4. CCSDS Primary Header Format for PDSS Payload CCSDS Packet

Byte	Bit	Field Name	Type	Values	Descriptions
0	7-5	Version	Binary	set to 000	Indicates CCSDS Version -1. Does not change.
	4	Type	Binary	Set to 0 if packet type is core (systems) data. Set to 1 if packet type is payload data.	Distinguishes between core (systems) and payload packet types to extend the APID space to 4064. For distribution external to the POIC, this will always be set to Payload.(1)
	3	Secondary Header Flag	Binary	Set to 0 if optional Secondary Header is not used. Set to 1 if optional Secondary Header is present	
	2-0	Application Process Identifier (APID) MSB	Integer	0-2031 (Decimal) (2032-2047 are reserved per CCSDS recommendation)	Used in conjunction with Type to define the specific source of USOS-generated Packets. Multiple data sources contained in a single physical location (e.g., separate payloads within an ISPR) must each be identified by at least one unique APID.
1	7-0	APID LSB			
2	7-6	Sequence Flags	Binary	‘00’B = Continuation Segment ‘01”B = First Segment ‘10’B = Second Segment ‘11’B = Unsegmented data set to 11 (Binary) (unused)	Always set to “11’B for USOS packets. Segmented Packets not supported by ISS Ku-Band Downlink. (Not used by POIC.)

Table 8-4. CCSDS Primary Header Format for PDSS Payload CCSDS Packet

Byte	Bit	Field Name	Type	Values	Descriptions
	5-0	Packet Sequence Count MSB	Integer	0-16383 (Decimal)	This 14-bit counter will be unique for each APID and incremented between successive packets belonging to the same APID. In ISS, these packets are numbered according to a Logical Data Path, (i.e., a separate counter is maintained for each source-destination pair.)
3	7-0	Packet Sequence Count LSB			
4	7-0	Packet Length MSB	Integer	94-4090 (Decimal)	This is the length of the data in bytes (octets) minus 1 that follows the CCSDS primary header.
5	7-0	Packet Length LSB			

8.2.3.2 CCSDS SECONDARY HEADER FORMAT FOR PDSS PAYLOAD CCSDS PACKET

The format for the CCSDS Primary Header format is consistent with Software Interface Control Document, Part 1, United States On-Orbit Segment (USOS) to International Ground System Segment Ku-Band Telemetry Formats (SSP 41158), Table 4.1.1-2 Secondary Header Field Definitions. These data are not modified by the POIC system.

Table 8-5. CCSDS Secondary Format for PDSS Payload CCSDS

Byte	Bit	Field Name	Type	Values	Descriptions
0-1	15-0	Two Most Significant Bytes of Coarse Time	Binary		CCSDS Unsegmented Time Code The Time code is made up of a preamble field (P-field) and a time field (T-field). For ISS, as allowed by the CCSDS Recommendation, the P-Field is implicitly conveyed (it is not present with the T-field) The T-field consists of 4 bytes of course time and 1 byte of fine time. These bytes are a set of binary counters, cascaded with the adjacent counters. The value represents the elapsed time since midnight 5-6 January 1980. The least significant bit (LSB) of the least significant byte of coarse time is equal to 1 second.
2-3	15-0	Two Least Significant Bytes of Coarse Time			
4	7-0	Fine Time	Binary		The LSB of the fine time byte is equal to 2^{-8} second, or about 4 ms.
5	7-6	Time ID	Binary	00 = Time Field Not Used 01 = Time Field Used	For Payload Data Packets, the field is set to 01 by the originating payload, if the payload has provided time information. For Communications Outage Recorder (COR) packets containing packetized bitstream data, the time field indicates when the bitstream data were packetized at the COR.

Table 8-5. CCSDS Secondary Format for PDSS Payload CCSDS

Byte	Bit	Field Name	Type	Values	Descriptions
	5	Checkword Indicator	Binary	Set to 0 - unused.	Unused by Ku-Band.
	4	ZOE TLM	Binary	Set to 0 - unused	Used by S-Band ZOE Packets contained in the Ku-Band Return link. Unused by Ku-Band.
	3-0	Packet Type	Binary	Set to 0000-unused	Unused by Ku-Band.
6-7	15-0	Version Identifier	Integer	1-65535	Specifies data field structure, using pre-defined Ku-Band telemetry data format defined in the ISS Payload Data Library (PDL).
8-9	15-0	Data Cycle Counter	Integer	1-65535	Identifies Packet as being Packet number "n" in a multi-packet data cycle.

8.2.3.3 CCSDS DATA ZONE FOR PDSS PAYLOAD CCSDS PACKET

The PDSS Payload Consultative Committee for Space Data Systems (CCSDS) Packet data are not modified by the POIC. The packet is simply reassembled and distributed. No data are added or modified from the original onboard content within the PDSS Payload CCSDS Packet. There are no limits or constraints on the data for it to be delivered to the remote user's GSE if the data are not processed by the POIC and just put through to the remote user's GSE.

8.3 PDSS PAYLOAD BITSTREAM-PROTOCOL DATA UNIT (BPDU)

PDSS Payload Bitstream Protocol Data Unit (BPDU) contains payload bitstream data downlinked from the ISS. The remote user can receive PDSS Payload BPDUs through the PDSS distribution of real-time data. In order to access this real-time distribution of data, the remote user must specify the requirement to access these telemetry data through the PDL requirement process as defined in section 1.3.

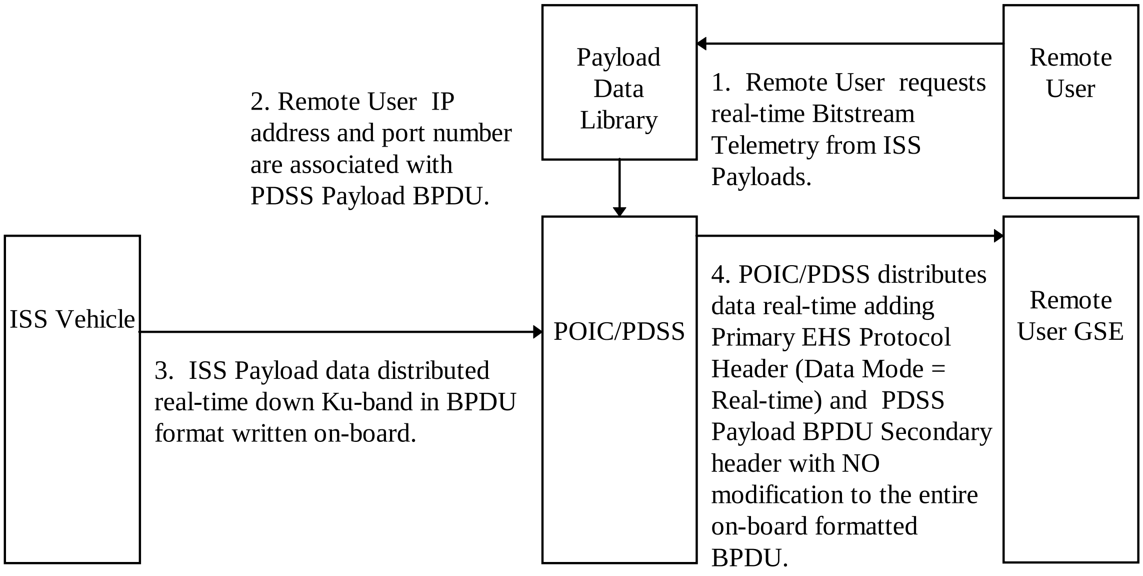

Payload Data Library
Remote User
1. Remote User requests real-time Bitstream Telemetry from ISS Payloads.
2. Remote User IP address and port number are associated with PDSS Payload BPDU.
ISS Vehicle
POIC/PDSS
Remote User GSE
3. ISS Payload data distributed real-time down Ku-band in BPDU format written on-board.
4. POIC/PDSS distributes data real-time adding Primary EHS Protocol Header (Data Mode = Real-time) and PDSS Payload BPDU Secondary header with NO modification to the entire on-board formatted BPDU.

Figure 8-6. Receiving PDSS Payload BPDUs

When the Secondary Protocol Header Type field in the Primary EHS Protocol Header is set to decimal value 13, the packet contains a PDSS BPDU. Figure 8-7 shows the EHS Protocol Data structure for PDSS BPDU. The EHS protocol data area contains a unique secondary protocol header definition followed by a BPDU.

Primary EHS Protocol Header (16 bytes)	Secondary EHS Protocol Header for PDSS BPDU (12 bytes)	BPDU (1094 bytes)
	EHS Protocol Data	

Figure 8-7. EHS Protocol Data Structure for PDSS BPDU

8.3.1 PRIMARY EHS PROTOCOL HEADER FOR PDSS BPDU

See section 8.1.1 for definition of the Primary EHS Protocol Header.

8.3.2 SECONDARY EHS PROTOCOL HEADER FOR PDSS BPDU

The Secondary EHS Protocol header for PDSS telemetry is 12 bytes (96 bits) in length. Table 8-6 defines the Secondary Protocol Header format for PDSS BPDU.

Table 8-6. Secondary EHS Protocol Header Format for PDSS Payload BPDU

Byte	Bit	Field Name	Type	Values	Description
0	7-6	Version Number	Binary	Set to 00	This is the PDSS secondary header version number. This number is independent of the EHS protocol version number in the primary header of the EHS protocol.
		Data Status			
	5-3	Unused	Binary	Set to 0	Unused
	2	VCDU Sequence Error	Binary	0=No error 1=Error occurred	Indicates if a VCDU Sequence Error was detected. Used to indicate that BPDU data may have been lost.
	1-0	Unused	Binary	Set to 0.	Unused
1-3	23-0	VCDU Sequence Number	Integer	Value set from VCDU Sequence Number from the VCDU Header	This is the VCDU sequence number taken from the VCDU header. This value is not changed by PDSS.
4	7	Data Stream Identifier (DSID) CCSDS vs. BPDU Identifier	Binary	1=BPDU	Fields from Bytes 4 -7 make up the Data Stream Identifier (DSID). This is a unique identifier for the data stream. This bit determines if the type of data contained therein is CCSDS packet data (0) or BPDU(1).
	6-0	Unused	Binary	Set to all zeros.	Reserved for future use.
5	7-0	Unused	Binary	Set to all zeros.	Reserved for future use.
6	7-5	Unused	Binary	Set to zeros.	Reserved for future use.

Table 8-6. Secondary EHS Protocol Header Format for PDSS Payload BPDU

Byte	Bit	Field Name	Type	Values	Description
	4	DSID GSE Packet Identifier	Binary	Set to zero.	Set to 0 to specify this is not a POIC created GSE packet.
	3	DSID Payload vs. Core Identifier	Binary	0=Core (Systems) data 1=Payload data	Corresponds to the type bit and is set to 1 for payload data and set to 0 for core (systems) data
	2-0	DSID APID MSB	Integer	0-2047 (Decimal)	The eleven bits correspond to the user's APID
7	7-0	DSID APID LSB			
8-9	15-0	Routing ID	Integer	PDSS internal routing key 0-65535 (Decimal)	This is a routing key used internal to PDSS. This value will be of a dynamic nature.
10-11	15-0	PDSS Reserved	Binary	Set to 0111 1011 1101 1110 (or 7BDE hex.)	This key is used internally to PDSS and should not be used by the remote user. This is a fixed sixteen-bit pattern.

8.3.3 BITSTREAM-PROTOCOL DATA UNIT (BPDU)

The BPDU contains its own header information that is specific to the telemetry data being transmitted. The format of the BPDU is defined in ISS United States On-Orbit Segment to Ground (Through Tracking and Data Relay Satellite System) Interface Control Document (SSP 42018), section 4.4.2.2, Bitstream Protocol Data Unit. The POIC does not add to or modify the BPDU header or data. These data are only reassembled and distributed in the BPDU format as they were generated on the onboard ISS vehicle.

Warning on Data Validity within BPDU

Please note that there is a known problem with the onboard Bitstream Service from ISS. The High Rate Frame Mux (HRFM) does not meet CCSDS Bitstream Service as defined in CCSDS 701.0-B-2. Based upon a previously approved exception, the user was advised to check the entire BPDU for valid data versus fill data whenever a Bitstream Data Pointer value of "07FF" was encountered. The fill pattern to check was based upon a project-specified fill pattern beginning with the hexadecimal sequence "BFC3FFF56". However, the HRFM does not meet the previously approved exception and is not using the project–specified Fill Pattern. Under nominal conditions, the HRFM will completely fill BPDUs with user data. The invalid Bitstream Data Pointer of "07FF" (hex) can safely be assumed to be for a BPDU full of user data (i.e., no fill data). However, when the user data really ends at a Data Pointer of "07FF" (hex), the Bitstream Data Pointer of "07FF" (hex) will incorrectly be determined to be for a BPDU full of user data. This will result in additional junk data being added to the end of the user data. This condition is rare, but it can occur when the user has finished sending data, or when the High Data Rate Link (HDRL) is allowed to timeout and the BPDU just happens to be filled with user data to a Data Pointer of exactly "07FF" (hex).

PDSS does not look into the BPDU to distinguish between valid data and fill data. PDSS will encapsulate the BPDUs using an Enhanced Huntsville Operations Support Center System (EHS) Header and pass this encapsulated data to the user regardless of this error. The user will have to process the data by removing the EHS Header and looking at the entire BPDU to determine this type of error.

The HRFM requires that Bitstream data be provided in 16–bit words. That is, the HRFM only builds BPDUs on 16–bit data word boundaries and does not point to bit level. For the 16–bit data word boundary problem, SSP 50184 (HRDL specification) has imposed the 16–bit boundary requirement on all users.

8.4 PDSS USER DATA SUMMARY MESSAGE (UDSM)

PDSS UDSMs represent data quality reported by Data Stream Identifier (DSID) for a particular user. There are three different UDSM events that cause UDSMs to be sent to the remote user. They are:

a. Scheduled Loss of Signal.

    (1) Loss of Signal based upon predicted Acquisition of Signal/Loss of Signal (AOS/LOS).

    (2) End of LOR/White Sands Complex (WSC) playback is treated as a Scheduled Loss of Signal.

b. Request for UDSM output by the PDSS operator.

c. End of a user’s data as specified by the payload mission/increment timeline.

The UDSM is for the packet or BPDU stream that the user is receiving from the ISS vehicle. The UDSM corresponds with the DSID. The data associated with the UDSM are based upon the Mission, Operational Support Mode, and Data Mode as specified in the EHS Primary Protocol Header. If the remote user is receiving the same data in two different Data Modes, then the remote user will receive a UDSM for each Data Mode and DSID combination.

UDSMs contain the user’s DSID, the UDSM event, the start and stop time of the report period, the number of dropouts for the report period, the number of CCSDS packets or BPDUs transmitted (excluding UDSMs), the number of Virtual Channel Data Unit (VCDU) sequence errors (bitstream data only), and the number of packet sequence errors (packet data only). The remote user automatically receives the UDSM data if the user has requested to receive PDSS Payload CCSDS Packet data or PDSS BPDUs. The UDSM will be sent to the same UDP IP and port address to which the source packet is delivered.

When the Secondary Protocol Header Type field in the Primary EHS Protocol Header is set to decimal value 14, the packet contains a PDSS UDSM. Figure 8-9, EHS Protocol Data Structure for PDSS UDSM, shows the EHS Protocol Data structure for PDSS UDSM. The EHS protocol data area contains a unique secondary protocol header definition followed by a CCSDS packet definition.

8.4.1 PRIMARY EHS PROTOCOL HEADER FOR PDSS UDSM PACKET

See section 8.1.1 for definition of the Primary EHS Protocol Header.

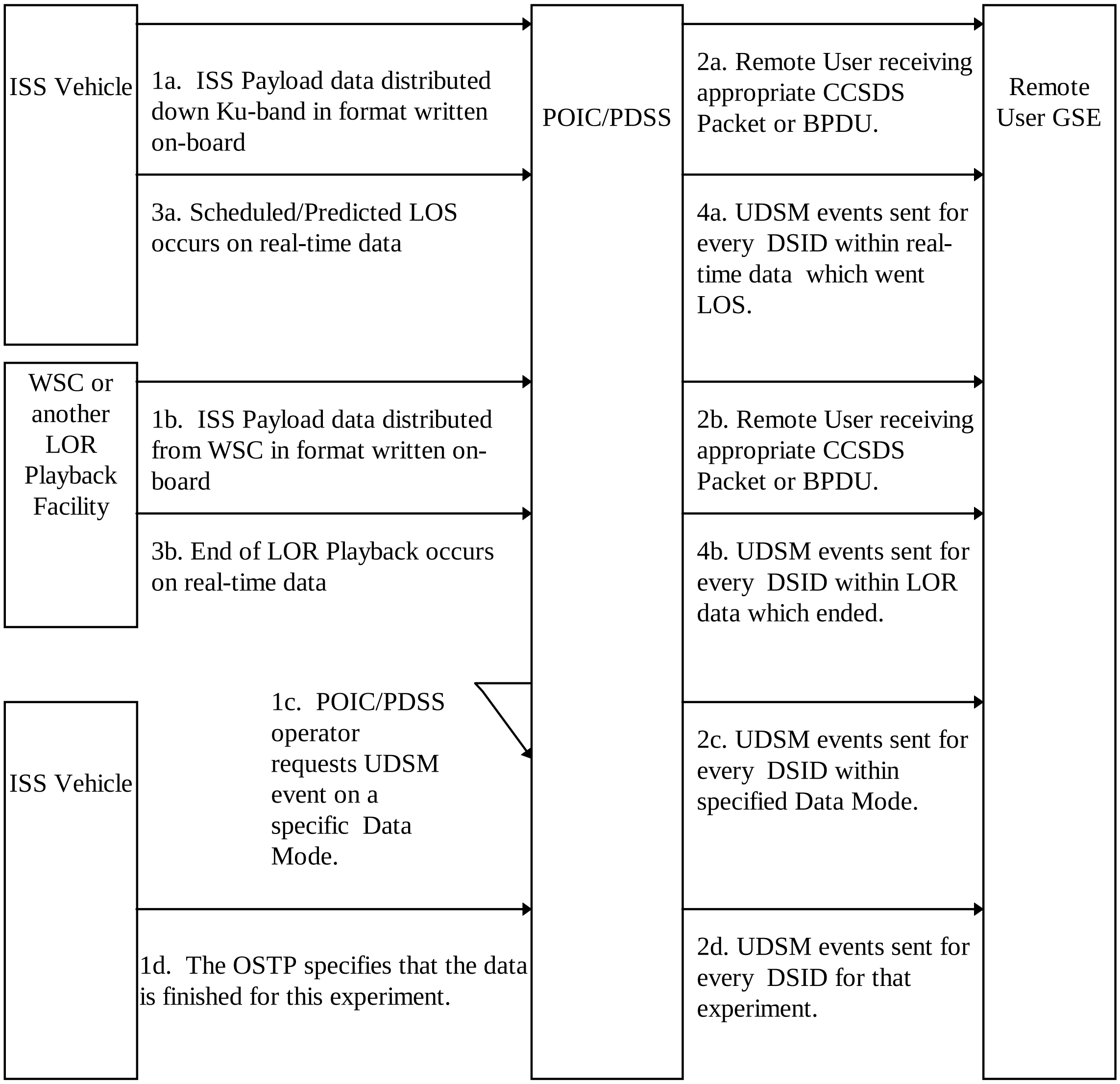

ISS Vehicle
1a. ISS Payload data distributed down Ku-band in format written on-board
POIC/PDSS
2a. Remote User receiving appropriate CCSDS Packet or BPDU.
Remote User GSE
3a. Scheduled/Predicted LOS occurs on real-time data
4a. UDSM events sent for every DSID within real-time data which went LOS.
WSC or another LOR Playback Facility
1b. ISS Payload data distributed from WSC in format written on-board
2b. Remote User receiving appropriate CCSDS Packet or BPDU.
3b. End of LOR Playback occurs on real-time data
4b. UDSM events sent for every DSID within LOR data which ended.
1c. POIC/PDSS operator requests UDSM event on a specific Data Mode.
2c. UDSM events sent for every DSID within specified Data Mode.
ISS Vehicle
1d. The OSTP specifies that the data is finished for this experiment.
2d. UDSM events sent for every DSID for that experiment.

Figure 8-8. Receiving UDSM Events

		CCSDS Packet	
Primary EHS Protocol Header (16 bytes)	Secondary EHS Protocol Header for PDSS UDSM (12 bytes)	CCSDS Primary Header (6 bytes)	UDSM (28 bytes)
	EHS Protocol Data		

Figure 8-9. EHS Protocol Data Structure for PDSS UDSM

8.4.2 SECONDARY EHS PROTOCOL HEADER

The Secondary EHS Protocol header for the PDSS UDSM is 12 bytes (96 bits) in length. Table 8-7 defines the Secondary Protocol Header format for PDSS UDSM.

Table 8-7. Secondary EHS Protocol Header Format for PDSS UDSM

Byte	Bit	Field Name	Type	Values	Description
0	7-6	Version Number	Binary	Set to 0	This is the PDSS secondary header version number. This number is independent of the EHS protocol version number in the primary header of the EHS protocol.
	5-0	Data Status	Binary	unused, Set to zeros	Does not apply to UDSMs.
1-3	23-0	VCDU Sequence Number		unused, Set to zeros	Does not apply to UDSMs.
4	7	Data Stream Identifier (DSID) CCSDS vs. BPDU Identifier	Binary	0= CCSDS 1=BPDU	Fields from Bytes 4 -7 make up the Data Stream Identifier (DSID). This is a unique identifier for the data stream. This bit determines if the type of data contained therein is CCSDS packet data (0) or BPDU(1).
	6-0	Unused	Binary	Set to all zeros.	Reserved for future use.
5	7-0	Unused	Binary	Set to all zeros.	Reserved for future use.
6	7-5	Unused	Binary	Set to zeros.	Reserved for future use.
	4	DSID GSE Packet Identifier	Binary	Set to zero.	Set to 0 to specify this is not a POIC created GSE packet.
	3	DSID Payload vs. Core Identifier	Binary	0=Core (Systems) data 1=Payload data	Corresponds to the type bit and is set to 1 for payload data and set to 0 for core (systems) data
	2-0	DSID APID MSB	Integer	0-2047 (Decimal)	The eleven bits correspond to the user's APID
7	7-0	DSID APID LSB			
8-9	15-0	Routing ID	Integer	PDSS internal routing key	This is a routing key used internal to PDSS. The PDSS operator will dynamically change this value.
10-11	15-0	PDSS Reserved	Binary	Set to 0111 1011 1101 1110 or 7BDE hex.	This key is used internally to PDSS and should not be used by the remote user. This is a fixed 16-bit pattern.

8.4.3 CCSDS PACKET FOR PDSS UDSM PACKET

The CCSDS packet is composed of a standard CCSDS Primary Header and a UDSM data zone.

8.4.3.1 CCSDS PACKET HEADERS FOR PDSS UDSM PACKET

See Table 8-8 for a definition of the CCSDS Primary Header. The APID and type bit in the CCSDS Primary Header will be set to match the data source that the UDSM represents. The Sequence count is set to zero to specify that the UDSM is not counting UDSMs since a separate UDSM is sent for each Data Mode and DSID combination. The CCSDS Secondary Header Flag is set to 0 for PDSS UDSM Packets to indicate that no secondary header is included in the CCSDS packets.

Table 8-8. CCSDS Primary Header for PDSS UDSM Packet

Byte	Bit	Field Name	Type	Values	Description
0	7-5	Version	Binary	Set to 000	Indicates CCSDS Version -1. Does not change.
	4	Type	Binary	Set to 1	Set to 1 if packet type is payload. A UDSM is considered part of the Payload data.
	3	CCSDS Secondary Header Flag	Binary	Set to 0	Secondary Header is not present
	2-0	Application Process Identifier (APID) MSB	Integer	0-2047 (Decimal)	The APID is set to match the data source that the UDSM represents.
1	7-0	APID LSB			
2	7-6	Sequence Flags	Binary	Set to 11 (Binary)	Not used by POIC.
	5-0	Packet Sequence Count MSB	Integer	Set to zero.	Not used for UDSMs. Each UDSM is unique and there are no sequences of data.
3	7-0	Packet Sequence Count LSB	Integer	Set to zero.	
4-5	15-0	Packet Length	Integer	0-60926 (Decimal) Set to 27	This is the length of the data in bytes (octets) minus 1 that follows the CCSDS primary header. A zero packet length indicates that the length of the data in bytes is either 0 or 1. In this case, the packet size from the EHS Primary Header should be used.

8.4.3.2 CCSDS DATA ZONE FOR PDSS UDSM PACKET

The format of the data in this packet is specified in Table 8-9 and will begin immediately after the CCSDS Primary Header.

Table 8-9. CCSDS Data Zone for PDSS UDSMs

Byte	Bit	Field Name	Type	Values	Description
0	7	Data Stream Identifier (DSID) CCSDS vs. BPDU Identifier	Binary	0= CCSDS 1=BPDU	Fields from Bytes 0-3 make up the Data Stream Identifier (DSID). This is a unique identifier for the data stream. This bit determines if the type of data contained therein is CCSDS packet data (0) or BPDU (1).
	6-0	Unused	Binary	Set to all zeros.	Reserved for future use.
1	7-0	Unused	Binary	Set to all zeros.	Reserved for future use.
2	7-5	Unused	Binary	Set to zeros.	Reserved for future use.
	4	DSID GSE Packet Identifier	Binary	Set to zero.	Set to 0 to specify this is not a POIC created GSE packet.
	3	DSID Payload vs. Core Identifier	Binary	0=Core (systems) data 1=Payload data	Corresponds to the type bit and is set to 1 for payload data and set to 0 for core (systems) data
	2-0	DSID APID MSB	Integer	0-2047 (Decimal)	The eleven bits correspond to the user's APID
3	7-0	DSID APID LSB			

Table 8-9. CCSDS Data Zone for PDSS UDSMs

Byte	Bit	Field Name	Type	Values	Description
4	7-0	Start Time Year	Integer	0-255 (Decimal)	The GRT Start Time for UDSM report period. Number of years since 1900
5-6	15-0	Start Time Day of Year	Integer	1-366 (Decimal), bits 15-9 are always 0	
7	7-0	Start Time Hours	Integer	0-23 (Decimal), bits 7-5 are always 0	
8	7-0	Start Time Minutes	Integer	0-59 (Decimal), bits 7-6 are always 0	
9	7-0	Start Time Seconds	Integer	0-61 (Decimal), bits 7-6 are always 0	
10	7-0	Stop Time Year	Integer	0-255 (Decimal)	The GRT Stop Time for UDSM report period. Number of years since 1900.
11-12	15-0	Stop Time Day of Year	Integer	1-366 (Decimal), bits 15-9 are always 0	
13	7-0	Stop Time Hours	Integer	0-23 (Decimal), bits 7-5 are always 0	
14	7-0	Stop Time Minutes	Integer	0-59 (Decimal), bits 7-6 are always 0	
15	7-0	Stop Time Seconds	Integer	0-61 (Decimal), bits 7-6 are always 0	
16-17	15-0	Number of Unplanned LOS Occurrences	Integer	0- 65535(Decimal)	The number of unplanned LOS occurrences during the report period.
18-19	15-0	Number of CCSDS Packets or BPDUs Transmitted	Integer	0-65536(Decimal)	The number of CCSDS Packets or BPDUs Transmitted during the report period.
20-21	15-0	Number of VCDU Sequence Errors	Integer	0-65535 (Decimal)	The number of VCDU Sequence Errors occurring during the report period. (N/A for CCSDS Packets)
22-23	15-0	Number of Packet Sequence Errors	Integer	0-65535(Decimal)	The number of Packet Sequence Errors occurring during the report period (N/A for Bitstream data)
24-25	15-0	Number of Packet Length Errors	Integer	0- 65535 (Decimal)	The number of Packet Length Errors occurring during the report period (N/A for Bitstream data)
26	7-0	UDSM Event	Binary	1=Scheduled LOS 2=Scheduled End of Data 3=Operator request	The event that triggered the UDSM generation.
27	7-0	Unused		Set to 0's	Reserved for future use.

8.5 PDSS RETRIEVAL PROCESSING SUMMARY MESSAGE (RPSM)

When the Secondary Protocol Header Type field in the Primary EHS Protocol Header is set to decimal value 15, the packet contains PDSS RPSM. Content and format of PDSS RPSM is TBD 8-1.

8.6 GROUND SUPPORT EQUIPMENT PACKET PROTOCOL

GSE packets contain data extracted from any of the data that are processed by the POIC and are defined in the POIC Telemetry database. Figure 8-10 depicts the process for exchanging GSE packet protocols. The remote user creates the GSE Packet through the Telemetry Database application. In order to use this capability, the sources for the extracted parameters must be compliant with MSFC-STD-1274 and MSFC-DOC-1949, and must reside in the POIC Telemetry Database.

The GSE packet is also the mechanism for the remote user to receive the Ground Ancillary data. The Ground Ancillary Data Set contains S-Band core (systems) data, Ku-Band payload health and status data, Ku-Band flight ancillary data, approved Ku-Band science and engineering data, Shuttle Operational Data (OD), and POIC computed data. Once the GSE Packet is defined in the Telemetry Database, the generic user can request the receipt of the GSE Packet through the GSE Packet application described in the POIC Capabilities Document (SSP 50304). The GSE Ancillary Data Set can contain more than one GSE packet depending upon how it is defined. The POIC Capabilities Document contains further information regarding GSE packets and Ground Ancillary GSE Packets capabilities in section 4. The GSE Packet is distributed using the UDP Protocol (See section 3.3.2.3.2 for more information on the UDP Protocol).

Figure 8-10. Exchanging GSE Packet Protocols

When the GSE Packet is being distributed to the generic user, it continues until the user requests that the distribution stop. If during the distribution of a GSE Packet an AOS to LOS transition occurs, the GSE Packet will be distributed with stale data (see Table 8-17 for the Status Characters Definition and when data received are stale or valid.)

When the Secondary Protocol Header Type field in the EHS Protocol Header is set to decimal 6, the packet contains GSE Packet telemetry. Figure 8-11 shows the EHS Protocol Data structure for GSE Packet telemetry. The maximum size of the GSE Packet is dependent upon whether it is distributed by PDSS. PDSS routed data are limited to 16,384 bytes for the entire packet (including all the headers). Since Ground Ancillary Data Sets are distributed by PDSS, they are limited to the 16K size. However, the user generated GSE Packet is limited to 61,438, which includes all headers (limit is 61,399 without headers).

NOTICE: GSE Packets are processed by the POIC's ERIS server, which also processes all Custom Data Packets (CDPs) and command services (see PGUIDD Volume 2). Together, these data requirements place a sizable load on ERIS resources. The POIC requests that users make efficient use of ERIS resources by expeditiously "stopping" GSE Packets when the associated data is not in use.

		CCSDS Packet (Maximum 61,406 bytes)		
Primary EHS Protocol Header (16 bytes)	Secondary EHS Protocol Header for GSE (16 bytes)	CCSDS Primary Header (6 bytes)	CCSDS Secondary Header (1 byte)	GSE Packet Data (Maximum Size 61,399 bytes for user generated GSE packets; 16,345 bytes for Ground Ancillary GSE packets distributed by PDSS.)
	EHS Protocol Data (Maximum 61422 bytes)			

Figure 8-11. EHS Protocol Data Structure for GSE Telemetry

8.6.1 PRIMARY EHS PROTOCOL HEADER FOR GSE PACKET

See section 8.1.1 for definition of the Primary EHS Protocol Header.

8.6.2 SECONDARY EHS PROTOCOL HEADER FOR GSE PACKET

The Secondary EHS Protocol header for PDSS telemetry is 16 bytes (128 bits) in length. Table 8-10 defines the Secondary Protocol Header format for GSE Packets. The Workstation ID, User ID, and Process ID fields in the Secondary EHS Protocol Header are meant to be used only by EHS software developers.

Note: Even though the Secondary EHS Protocol Header format contains a variable called DSID Payload vs. Core Identifier, this flag is always set by the PDSS at the POIC to "Payload". However, GSE packets can include any combination of core, payload, and POIC-generated pseudo-telemetry data (including AOS/LOS indicators).

Table 8-10. Secondary EHS Protocol Header Format for GSE Packets

Byte	Bit	Field Name	Type	Values	Description
0	7	Packet Routing Flag	Binary	0 = TNS Routed 1 = PDSS Routed	Bit set to PDSS Routed to distinguish which GSE Packets are distributed by PDSS.
	6	User Interface Request Type	Binary	0 = POIC 1 = Remote user	
	5-0	Unused	Binary	Set to 0.	
1	0-7	Unused	Binary	Set to 0.	
2-3	15-0	Workstation ID	Integer	0-65535 (Decimal)	Set to the Workstation numerical ID that is running the GSE process that created the GSE packet
4	7	Data Stream Identifier (DSID) CCSDS vs. BPDU Identifier	Binary	0= CCSDS Packet 1=BPDU Always set to zero.	Fields from Bytes 4 -7 make up the Data Stream Identifier (DSID). This is a unique identifier for the data stream. This bit determines if the type of data contained therein is CCSDS packet data or BPDU
	6-0	Unused		Set to all zeros.	Reserved for future use.
5	7-0	Unused		Set to all zeros.	Reserved for future use.
6	7-5	Unused		Set to zeros.	Reserved for future use.
	4	DSID GSE Packet Identifier	Binary	Set to 1.	Set to 1 to specify this is a POIC created GSE packet.
	3	DSID Payload vs. Core Identifier	Binary	0=Core (Systems) data 1=Payload data Always set to 1.	Corresponds to the type bit and is set to 1 for payload data and set to 0 for core data.
	2-0	DSID APID MSB	integer	0-2047 (Decimal)	The eleven bits correspond to the user's APID.
7	7-0	DSID APID LSB			
8-9	15-0	Routing ID		PDSS internal routing key	For Ground Ancillary GSE Packets: This is a routing key used internal to PDSS. This value will be of a dynamic nature. For User-Generated GSE Packets: nonused. Set to 0's.
10-11	15-0	PDSS Reserved	Binary	Set to 0111 1011 1101 1110 (or 7BDE hex)	This key is used internally to PDSS and should not be used by the remote user. This is a fixed 16-bit pattern.
12-13	15-0	User ID	Integer	0-65535 (Decimal)	Numerical internal representation of the User ID that executed the GSE process
14-15	15-0	Process ID	Integer	0-65535 (Decimal)	Numerical internal representation of the process ID of the GSE process.

8.6.3 CCSDS PACKET FOR GSE PACKET

The CCSDS packet is composed of a standard CCSDS Primary Header, a CCSDS Secondary Header, and a GSE Packet data zone.

8.6.3.1 CCSDS PACKET HEADERS FOR GSE PACKET

See Table 8-11 for a definition of the CCSDS Packet format and a definition of the CCSDS Primary Header. The APID in the CCSDS Primary Header is assigned according to the GSE Packet Definition as specified in the GSE Packet Definition File (See section 12.3 for more details on the GSE Packet Definition File from the Telemetry Database). The GSE Packet ID is used as an APID in the GSE Packet. The GSE Packet IDs are preassigned to the generic user through PDL and thus are unique within the GSE protocol. The POIC GSE Packet Definition File input restricts the user to these preassigned GSE Packet IDs. The combination of the APID and the GSE Packet type (the latter is set in the Secondary Protocol Header Type field in the Primary EHS Protocol Header) guarantee a unique identifier across all packet types. The Packet Sequence counter will be incremented for each GSE Packet that corresponds with the same APID and MOP combination that occurs after the GSE Packet has started transmission. If the remote user stops the transmission of the GSE Packet and restarts it, the Packet Sequence Counter will reset to 0.

Table 8-11. CCSDS Primary Header for GSE Packet

Byte	Bit	Field Name	Type	Values	Description
0	7-5	Version	Binary	Set to 000	Indicates CCSDS Version -1. Does not change.
	4	Type	Binary	Set to 1	Set to 1 if packet type is payload. A GSE Packet is considered part of the Payload data.
	3	CCSDS Secondary Header Flag	Binary	Set to 1	Secondary Header is present
	2-0	Application Process Identifier (APID) MSB	Integer	0-2047 (Decimal)	The APID is set to the APID value assigned in the POIC Telemetry Database for this GSE Packet.
1	7-0	APID LSB			
2	7-6	Sequence Flags	Binary	Set to 11 (Binary)	Not used by POIC.
	5-0	Packet Sequence Count MSB	Integer	0-16383 (Decimal)	This 14-bit counter will be unique for each APID/MOP combination and will be incremented between successive packets belonging to the same APID.
3	7-0	Packet Sequence Count LSB	Integer		
4-5	15-0	Packet Length	Integer	0-61399 (Decimal)	This is the length of the data in bytes (octets) minus 1 that follows the CCSDS primary header. A zero packet length indicates that the length of the data in bytes is either 0 or 1. In this case, the packet size from the EHS Primary Header should be used.

8.6.3.2 CCSDS SECONDARY HEADER FOR GSE PACKET

A GSE Packet contains a CCSDS Secondary Header that identifies packet format and determines whether the Select Merge Processing option will be used to generate the GSE Packet from Near Real Time (NRT) data storage.

The format of a GSE Packet includes the packet's update cycle, sampling sequence, and measurement set. The Format ID in the CCSDS Secondary Header identifies which pre-defined format is associated with the GSE packet. The detailed format information associated with each Format ID (0 thru 7) is defined in the POIC Telemetry Database.

The CCSDS Secondary Header for a GSE Packet also includes a "Select Merge Processing Flag". The Select Merge Processing Flag provides remote users with a simple way to request Near Real Time (NRT) playback telemetry data directly from their "GSE Packets" web application. From within the GSE Packets web application the user can choose the "Select Merge Data" data mode option, which sets the Select Merge Processing Flag bit to '1' ("on") in the CCSDS Secondary Header. The user can then request that data from their GSE packet be played back to them for a specified time period from the past, e.g., if a significant event occurred one month ago that the user would like to see again, the user can specify the dates surrounding that event and will receive the desired NRT playback.

The Select Merge Processing Flag bit can be ignored, but the user will lose a simple method for receiving historical data. If Select Merge Data is not used, the only other method for requesting Near Real Time playbacks from the user's web interface is to use the "Playback" option (available beginning with EHS Build 5.0) within the GSE Packets application. The Playback option also allows the user to request NRT data for a user-defined time period, but requires the user to know the detailed contents of their GSE packet so the desired data stream can be reconstructed from the NRT archives. The Select Merge Data option is easier to user because it only requires knowledge of the GSE packet ID to request NRT data for the desired timeframe.

Table 8-12. CCSDS Secondary Header for GSE Packet

Byte	Bit	Field Name	Type	Values	Description
0	7-5	Format ID	Integer	0-7 (Decimal)	Format ID is set to value assigned in Telemetry Database.
	4	Select Merge Processing Flag	Binary	0 = off 1 = on	Indicates whether Select Merge Processing was used to generate GSE packet.
	3-0	Unused	Binary	Set to 0.	

8.6.3.3 CCSDS DATA ZONE FOR GSE PACKET

The GSE Packet uses the Telemetry Database to determine the order and maximum number of sample size associated with each MSID. This will also determine the size of the GSE Packet. The data associated with each MSID are delineated in Table 8-14. The information is repeated for each MSID in the GSE Packet in the order of the MSIDs as specified in the Telemetry

Database. The GSE Packet Definition file format that is retrieved from the Telemetry Database is defined in section 12.3.

The time associated with the data in the GSE Packet is contained in the Primary EHS Protocol Header in the Ground Receipt Time field. Depending upon which time type is requested by the user on the GSE Packet application, this time will be either the POIC creation time or embedded MSID time. The POIC creation time is the time (either real-time or simulation time) that the packet is created. The embedded MSID time is the time MSID selected from a particular source packet by the remote user. If the GSE Packet is made up of two different source packets and the remote user requests embedded MSID time, then the remote user must choose which embedded MSID time should be distributed with the GSE Packet. It is prudent to limit the user-generated GSE Packet to only one source packet. The remote user is not limited to the number of GSE Packets that can be routed to the remote user's GSE.

The GSE Packet contains all the available information associated with the desired MSIDs for the timeframe specified for the GSE Packet. Some MSIDs will have multiple samples associated with the timeframe since the type of MSID can receive multiple values. The Telemetry Database will have defined the maximum number of samples that can be received within a "cycle" for that MSID. The values in Table 8-14 are repeated for each MSID received in the GSE Packet. To determine the offset between the MSIDs, the MSID Start Octet value from the Telemetry Database is used to determine the starting position for a specific MSID within that GSE Packet (see Figure 8-12).

Note: The GSE Packet data zone does NOT contain the name of the MSID. That information is available in the Telemetry Database.

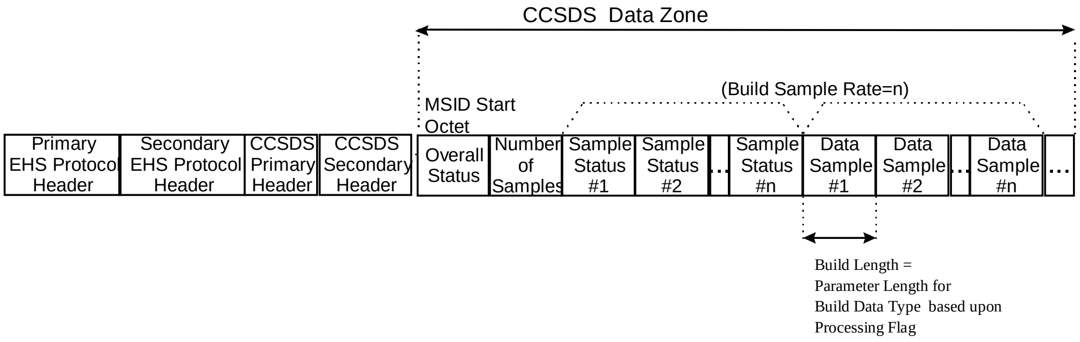
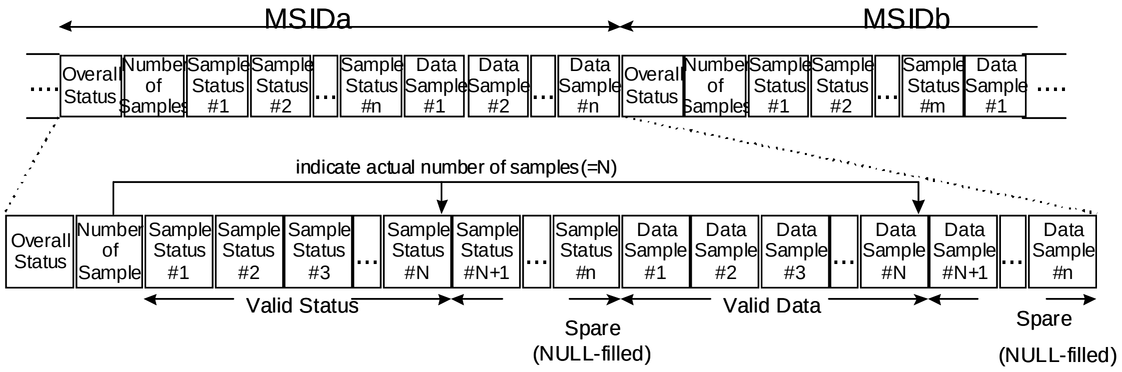


Figure 8-12. MSID Data Position Within CCSDS Data Zone

In Table 8-13 the Overall Status starts the values for the MSID. For more information on the Overall Status value see Tables 8-15 and 8-16. The Number of Samples in Table 8-14 specifies the actual number of samples that are associated with this GSE Packet for this MSID. For example, if "MSIDa" in the Telemetry Database has a maximum number of 10 samples, yet for this timeframe only four samples were collected, the Number of Samples would be four. The Telemetry Database GSE Packet Definition File defines the maximum number of samples with the Build Sample Rate variable (see section 12.3).



N: actual number of samples of MSIDa
n: Build Sample Rate for MSIDa or
maximum number of samples of MSIDa
m: Build Sample Rate for MSIDb or
maximum number of samples of MSIDb

Figure 8-13. Example Parameter Information within CDP and GSE Packets

The Status Character for the samples is next in the format of the GSE Packet. In the example, this would mean that you have 4 status characters. Now, since in the Telemetry Database "MSIDa" is defined as having a maximum of 10 samples, the next 6 (i.e., ten minus four) status characters would be zero-filled. The status character priority is defined in Table 8-17.

Note: When the Number of Samples field is 0, the first Sample Status field will contain a status character corresponding to the Overall Status. The remaining Sample Status fields and all of the Data Sample fields are zero-filled.

Next would come the four values for MSIDa (see Figure 8-13). Once again, since in the Telemetry Database "MSIDa" is defined as having a maximum of 10 samples, the next 6 samples would be zero-filled. At this point, the information associated with MSIDa is finished for this GSE Packet, and the next MSID would then start with the Overall Status and reiterate the values for that MSID.

The type of values that are contained in the samples and the type of processing performed against those values are both contained in the Telemetry Database. Thus, even though the size of the sample as defined in Table 8-14 is called out as only one byte, the actual size of the sample would have to be determined from the definition in the Telemetry Database. The processing types possible for an MSID in a GSE Packet are unprocessed, converted, and calibrated. Table 10-1 delineates the formats for these different data types in unprocessed, converted, and calibrated forms.

Data in a GSE Packet are inserted on a byte boundary. For example, if the data contain two unprocessed samples of a 3-bit discrete integer, the data would be placed in two bytes (with each byte right justified) in the GSE Packet, as specified in Table 8-13.

Table 8-13. GSE/CDP Data Zone Unprocessed Data Type Justification

Data Type	Justification	Fill value
String	Left	NULL (x0)
Integers	Right	Zeros
Floats	Right	zeros
Times	Right	zeros

The GSE Packet is sent to the IP address and port specified by the remote user. If multiple GSE Packets are being requested by the remote user, they may all go to the same port address or separate port addresses.

Table 8-14. CCSDS Data Zone for GSE and CDP Packets

Byte	Bit	Field Name	Type	Values	Description
0-3	31-0	Overall Status	Integer	See Table 8-14 and Table 8-15.	
4-7	31-0	Number of Samples (N)	unsigned integer	0-255 (Decimal), right justified zero filled	
8	7-0	Sample Status	char	See Table 8-16 for valid Values for status characters	
....	...	*Sample status*	*char*	...	*Sample status repeats for number of samples*
8+N		Sample Status Spare Status Fields	char	Set to (NULL (x0).	Sample Status spare status field repeats until the Maximum Samples (M) is reached. Maximum Samples is specified in Telemetry Database.
8 + M		Data sample		Data type and value as specified in Telemetry Database	
...	...	*Data Sample*		Data type and value as specified in Telemetry Database	*Data sample repeats for number of samples*
8 + M+N	...	Spare from Maximum Number of Samples		Start Byte (set to NULL (x0))	
...	...	*Spare from Maximum Number of Samples*		*Set to NULL (x0)*	*Spare from Maximum Number of Samples -repeats until Max Samples is reached (set to NULL)*

Table 8-15 lists the encoded status portion of the Overall Status. The mask definition should be used to 'mask-off' other portions of the Overall Status when determining the encoded errors.

Table 8-15. Encoded Status of Overall Status

Telemetry Encoded Error Mask	0x000000FF
OK	0x00000000
STATUS_TABLE_ATTACH_ERROR	0x00000001
INIT_SOURCES_ERROR	0x00000002
INIT_PACKETS_ERROR	0x00000003
CDD_REQUEST_ERROR	0x00000004
LOCAL_TABLE_ERROR	0x00000005
MEMORY_ERROR	0x00000006
PACKET_INTERRUPT_ERROR	0x00000007
REQUEST_UPDATES_ERROR	0x00000008
SMAC_TDS_ERROR	0x00000009
DP_RECONFIG_ERROR	0x0000000A
PACKET_NOT_FOUND	0x0000000B
DP_ADDRESS_ERROR	0x0000000C
DP_PID_ERROR	0x0000000D
NPA_FIFO_ERROR	0x0000000E
MAX_PACKET_LENGTH_ERROR	0x0000000F
DECOM_ERROR	0x00000010
INVALID_DATA_TYPE	0x00000011
SW_CONV_ERROR	0x00000012
INVALID_LENGTH	0x00000013
IMPROPER_FUNCTION_TYPE	0x00000014
INVALID_SAMPLE_TYPE	0x00000015
NO_DECOM_SAMPLES	0x00000016
PLVT_ATTACH_ERROR	0x00000017
LVT_ATTACH_ERROR	0x00000018
PIT_ATTACH_ERROR	0x00000019
NEW_LOCAL_TABLE	0x0000001A
SOURCE_UNAVAILABLE	0x0000001B
PDB_ATTACH_ERROR	0x0000001C
CLEANUP_SOURCES_ERROR	0x0000001D
SOURCE_NOT_FOUND	0x0000001E
SCM_STATUS_DETACH	0x0000001F
PIT_DETACH_ERROR	0x00000020
REMOVE_PACKET_ERROR	0x00000021
STATUS_TABLE_DETACH_ERROR	0x00000022
PIT_REVISION_COUNTER_ERROR	0x00000023
LOCAL_TABLE_REVISION_COUNTER_ERROR	0x00000024
PIT_ERROR	0x00000025
REMOVE_PACKET_UPDATES_ERROR	0x00000026
SEMID_ERROR	0x00000027
FTOK_ERROR	0x00000028
PARAMETER_DATABASE_ERROR	0x00000029

Table 8-15. Encoded Status of Overall Status

SHM_KEY_ERROR	0x0000002A
CONVERT_ERROR	0x0000002B
CAL_ERROR	0x0000002C
LES_ERROR	0x0000002D
SWITCH_CAL_ERROR	0x0000002E
SWITCH_CONVERT_ERROR	0x0000002F
SWITCH_DECOM_ERROR	0x00000030
SEM_ACQUIRE_ERROR	0x00000031
PDB_ID_ERROR	0x00000032
SEM_RELEASE_ERROR	0x00000033
TDS_PACKET_QUESTIONABLE	0x00000034
GET_MSID_VALUE_ERROR	0x00000035
DECREMENT_SPC_ERROR	0x00000036
UPDATE_STATUS_TABLE_ERROR	0x00000037
INVALID_FORMAT	0x00000038
TIME_NOT_AVAILABLE	0x00000039
INIT_NEW_SOURCE_ERROR	0x0000003A
CAL_SETS_UNDEFINED	0x0000003B
LIMIT_SETS_UNDEFINED	0x0000003C
PDB_DETACH_ERROR	0x0000003D
NO_LOCAL_TABLE_INFO	0x0000003E
INVALID_FUNCTIONS	0x0000003F
NO_MSID_DATA_MEMORY	0x00000040
INCREMENT_SPC_ERROR	0x00000041
EXPECTED_SAMPLE_SIZE_ERROR	0x00000042
DATA_TYPE_LES_MISMATCH	0x00000043
DATA_TYPE_CAL_TYPE_MISMATCH	0x00000044
MJFB_ATTACH_ERROR	0x00000045
MJFB_DETACH_ERROR	0x00000046
TNS_KEY_FILE_PATH_ERROR	0x00000047
CREATE_PACKET_NOTIFY_KEYFILE_ERROR	0x00000048
CREATE_PACKET_NOTIFY_SEMKEY_ERROR	0x00000049
CREATE_PACKET_NOTIFY_SEMID_ERROR	0x0000004A
KILL_PACKET_NOTIFY_SEMID_ERROR	0x0000004B
PACKET_LIST_ERROR	0x0000004C
INVALID_CONTEXT_TYPE	0x0000004D
CDD_SERVICE_OPEN_ERROR	0x0000004E
CDD_SERVICE_CONNECT_ERROR	0x0000004F
CDD_SERVICE_SEND_ERROR	0x00000050
CREATE_LVT_SEMKEY_ERROR	0x00000051
GET_LVT_SHMID_ERROR	0x00000052
GET_LVT_SEMID_ERROR	0x00000053
LVT_DETACH_ERROR	0x00000054

Table 8-15. Encoded Status of Overall Status

PLVT_DETACH_ERROR	0x00000055
MSID_NOT_INITIALIZED	0x00000056
SOURCE_INFO_ERROR	0x00000057
UPDATE_LOCAL_TABLE_ERROR	0x00000058
SET_LES_PROC_ERROR	0x00000059
UPDATE_LTRC_ERROR	0x0000005A
UIT_CONNECT_ERROR	0x0000005B
UIT_DISCONNECT_ERROR	0x0000005C
SET_LES_PROC_FLAG_ERROR	0x0000005D
INVALID_MAX_SAMPLES	0x0000005E
ACQUIRE_PDB_READ	0x0000005F
RELEASE_PDB_ERROR	0x00000060
OPEN_PDB_LOCK_FD	0x00000061
UITM_CONNECT_ERROR	0x00000062
GAM_REQUEST_ERROR	0x00000063
DATA_LOST	0x00000064
CDD_SERVICE_RECEIVE_ERROR	0x00000065
CONTEXT_SOURCE_ERROR	0x00000066
RANGE_NO_CAL_INFO	0x00000067
CONTEXT_SAMPLE_SIZE_ERROR	0x00000068
RANGE_CAL_ERROR	0x00000069
RANGE_CONVERT_ERROR	0x0000006A
CONTEXT_DECOM_ERROR	0x0000006B
CONTEXT_FORMAT_ERROR	0x0000006C

Table 8-16 lists the Non-Encoded Bit Definition portion of the Overall Status. The mask definitions should be used to ‘mask-off’ other portions of the Overall Status when determining the bit definitions.

Table 8-16. Non-Encoded Bit Definition of the Overall Status

Data Condition Mask	0x40000000
DC_NEW_DATA	0x00000000
DC_OLD_DATA	0x40000000
Data Source Mask	0x30000000
SS_INITIALIZED	0x20000000
SS_LOS	0x10000000
SS_AOS	0x00000000
Data Quality Mask	0x0F000000
DQ_SUSPECT	0x08000000
DQ_FAILED	0x04000000
DQ_NO_DATA	0x02000000
DQ_OVERRIDE	0x01000000

Table 8-16. Non-Encoded Bit Definition of the Overall Status

GSE Application Error Mask	0x00000F00
GSE_NUMBER_SAMPLES_OVERFLOW	0x00000100
GSE_DATA_TYPE_MISMATCH	0x00000200
Other Bit Definitions:	
PS_LENGTH_ERROR	0x00800000
PS_SEQUENCE_COUNT_ERROR	0x00400000
LES_OVERALL_STATUS_WORD_ERROR	0x00200000
RECONFIG_DETECTED	0x00100000
MJF_FORMAT_ID_ERROR	0x00080000
MJF_PARENT_FRAME_ERROR	0x00040000
PS_DATA_LOST	0x00020000

Table 8-17. Status Characters (Char) Definition

Status Char	Definition	Status Char Validity of Data	Priority of Status Char
‘ ‘ (space)	OK, Source Status - Acquisition Of Signal (AOS)	Valid	25
‘G’	Packet Routing Table Configuration Error	Not Valid	24
‘R’	Source Status - Source Initialized/ Unavailable	Stale Data	23
‘N’	Source Status - Loss Of Signal (LOS)	Stale Data	22
‘S’	Data Condition - Old/Stale Data	Stale Data	21
‘~’	Data Quality - No Data	Stale Data	20
‘F’	Data Quality - DQ Failed w/Override	Valid	19
‘?’	Data Quality - DQ Suspect w/Override	Valid	18
‘f’	Data Quality - DQ Failed	Stale Data	17
‘x’	Data Quality - DQ Is Suspect	Valid	16
‘Q’	Format ID Error	Stale Data	15
‘K’	Parent Frame Error	Stale Data	14
‘D’	Decom/Conv/Cal Status - Decom Error	Stale Data	13
‘c’	Decom/Conv/Cal Status – Calibration Error	Stale Data	12
‘C’	Decom/Conv/Cal Status - Conversion Error	Stale Data	11
‘&’	Calibration sets not defined in Local Table	Stale Data	10
‘l’	Limit/Expected State Sensing Error	Stale Data`	9
‘d’	Limits sets Not Defined In The Local Table	Valid	8
‘t’	Telemetry Local Table Error	Stale Data	7
‘H’	LES Status - Warning High	Valid	6
‘>’	LES Status - Caution High	Valid	5
‘<’	LES Status - Caution Low	Valid	4
‘L’	LES Status - Warning Low	Valid	3
‘E’	LES Status - Out Of Expected State	Valid	2
‘$’	Invalid Status Received From EML	Stale Data	1

When sample data are retrieved by the POIC, an overall status value is used to reflect any errors the POIC may have encountered obtaining all of the data samples. Each data sample also has an associated status value. By combining the overall status value and the sample value, a status character value can be determined. Table 8-18 provides a description of the possible status character values and their order of precedence. A status character other than ' ' (OK) indicates that an error occurred obtaining the data sample.

Note: If the Overall Status field contains the GSE_NUMBER_SAMPLES_OVERFLOW error, this means that the number of available samples for this MSID is greater than the maximum size specified in the GSE packet definition. In this case, the amount of sample data associated with this MSID will be dictated by the GSE packet definition. If the GSE_DATA_TYPE_MISMATCH error is found, this means that the data type for this MSID differs from the data type specified in the GSE packet definition. In this case, all fields following the Overall Status will be zero-filled, except for the first Sample Status field, which will contain the status character "T" (Telemetry Database Discrepancies).

If an MSID fails POIC data initialization, GSE will be unable to retrieve sample data associated with this MSID. In these cases, all data fields associated with this MSID will be NULL-filled, except for the overall status field and the first status character field. The overall status field will contain the initialization error code, and the status character field will contain one of the values found in Table 8-18.

Table 8-18. Status Character Definitions for MSID Initialization Failures

Status Character	Definition	Status Char Validity of Data	Priority of Status Char
'G'	Packet Routing Table Configuration Error	Not Valid	11
'I'	UITM Process Connect Error	Not Valid	10
'B'	Common Configuration Error	Not Valid	9
'W'	Memory Error	Not Valid	8
'A'	Distribute Packets (DP) Process Not responding	Not Valid	7
'a'	Context Dependent Decom (CDD) Process Not Responding	Not Valid	6
'T'	Telemetry Database Discrepancies	Not Valid	5
'&'	Calibration Sets Not Defined In Local Table	Not Valid	4
'P'	Telemetry Processing Discrepancies and Calibration Sets Not Defined in Local Table	Not Valid	3
'g'	Group Activation Manager (GAM) process Not Responding	Not Valid	2
'Z'	Unrecognized initialization Decom Status	Not Valid	1

Since each status character is assigned in order of precedence, only a single error condition can exist at one time. For example, if the overall status/sample status combination indicates than an MSID's source is initialized but unavailable, a status character of 'R' (Source Initialized But Unavailable) is assigned to the sample, and any other errors associated with the data sample will not be recognized.

8.7 CUSTOM DATA PACKET PROTOCOL

The CDP contains data extracted from any of the data that are processed in the POIC and are defined in the POIC Telemetry database. The CDP is distributed using TCP/IP protocol (see section 3.3.2.3.1 for more details about this protocol). The user determines the contents of a CDP using the request mechanism interface defined in section 10, Real-time Telemetry Requests, and the ERIS Interface defined in section 7.0. In order to use this capability, the sources for the extracted parameters must be compliant with MSFC-STD-1274 and MSFC-DOC-1949, and must reside in the POIC Telemetry Database. The POIC Capabilities Document (SSP 50304), section 4, contains further information regarding CDP capabilities.

When the Secondary Protocol Header Type field in the Primary EHS Protocol Header is set to decimal value 7, the packet contains CDP telemetry. Figure 8-14, EHS Protocol Data Structure for Custom Data Packets, shows the EHS Protocol Data structure for CDP telemetry. The EHS protocol data area contains a unique secondary protocol header definition followed by a CCSDS packet definition.

<table>
<tr><td></td><td></td><td colspan="2">CCSDS Packet (Maximum 61,406 Bytes)</td></tr>
<tr><td>Primary EHS Protocol Header (16 bytes)</td><td>Secondary EHS Protocol Header for CDP (16 bytes)</td><td>CCSDS Primary Header (6 bytes)</td><td>Custom Data Packet Data (same structure as GSE Packet Data; maximum size of 61,400 bytes)</td></tr>
<tr><td></td><td colspan="3">EHS Protocol Data</td></tr>
</table>

Figure 8-14. EHS Protocol Data Structure for Custom Data Packets

8.7.1 PRIMARY EHS PROTOCOL HEADER FOR CUSTOM DATA PACKETS

See section 8.1.1 for a definition of the Primary EHS Protocol Header.

8.7.2 SECONDARY EHS PROTOCOL HEADER FOR CUSTOM DATA PACKETS

The Secondary EHS Protocol header for CDPs shall be 16 bytes (128 bits) in length. Table 8-19 defines the Secondary Protocol Header format for CDPs. The Workstation ID, User ID, and Process ID fields in the Secondary EHS Protocol Header are used by CDP developers.

Table 8-19. Secondary EHS Protocol Header Format for CDP

Byte	Bit	Field Name	Type	Values	Description
0-1	15-0	Unused	Binary	Set to all zeros.	Reserved for future use.
2-3	15-0	Workstation ID	Integer	0-65535 (Decimal)	Set to the Workstation numerical ID that is running the CDP process.
4-11		Unused	Binary	Set to all zeros	Reserved for future use.
12-13	15-0	User ID	Integer	0-65535 (Decimal)	Numerical internal representation of the User ID that executed the CDP process
14-15	15-0	Process ID	Integer	0-65535 (Decimal)	Numerical internal representation of the process ID of the CDP process.

8.7.3 CCSDS PACKET FOR CUSTOM DATA PACKET

The CCSDS packet is composed of a standard CCSDS Primary Header and CDP data. The format of the data in this CCSDS packet is compliant with MSFC-STD-1274, Volume 2.

8.7.3.1 CCSDS PACKET HEADERS FOR CUSTOM DATA PACKETS

See Table 8-20 for a definition of the CCSDS Primary Header. The CCSDS Secondary Header Flag is set to 0 for CDPs. The APID assigned in the CCSDS Primary Header corresponds to the unique identifier returned to the requesting application in the CDP_Configure_Reply message (section 10.1.1). As long as the remote user process continues with that same CDP process session, the APID will not change. However, if the remote user stops the CDP process and performs a subsequent start or starts another CDP process, then a different APID number may be assigned. The APIDs will only be unique for all the CDP processes running at that moment in time for that UserID.

The packet sequence number is incremented for every CDP that is distributed for the specified APID. Every time a CDP_Configure request is received at the CDP Process, the Packet Sequence Number will be reset to zero.

Table 8-20. CCSDS Primary Header for Custom Data Packet

Byte	Bit	Field Name	Type	Values	Description
0	7-5	Version	Binary	Set to 000	Indicates CCSDS Version -1. Does not change.
	4	Type	Binary	Set to 1	Set to 1 if packet type is payload. A Custom Data Packet is considered part of the Payload data.
	3	CCSDS Secondary Header Flag	Binary	Set to 0	Secondary Header is not present
	2-0	Application Process Identifier (APID) MSB	Integer	0-2047 (Decimal)	The APID is set to the APID value returned in the CDP_Configure_Reply message (Section 10.2).
1	7-0	APID LSB			
2	7-6	Sequence Flags	Binary	Set to 11 (Binary)	Not used by POIC.
	5-0	Packet Sequence Count MSB	Integer	0-16383 (Decimal)	This 14-bit counter will be unique for each APID/MOP combination and will be incremented between successive packets belonging to the same APID.
3	7-0	Packet Sequence Count LSB	Integer		
4-5	15-0	Packet Length	Integer	0-61399 (Decimal)	This is the length of the data in bytes (octets) minus 1 that follows the CCSDS primary header. A zero packet length indicates that the length of the data in bytes is either 0 or 1. In this case, the packet size from the EHS Primary Header should be used.

8.7.3.2 CCSDS DATA ZONE FOR CUSTOM DATA PACKETS

The CCSDS Data Zone for the CDPs is identified in Table 8-14. The information in Table 8-14 is repeated for every MSID that is valid from the CDP Request (see section 10.1.1 for how to request a CDP). The data portion of the packet consists of an overall status for that MSID indicating whether any of the samples of that MSID contained in the packet had any errors or limit/expected state violations. The size of the packet is dependent upon the number of MSIDs and samples for each. The data do not specify the bit location since all data are a multiple of a byte size. Note that the CCSDS Data Zone for both GSE and CDP protocols are identical.

The CDP uses the CDP Configure message to determine the order of the MSIDs in the Custom Data Packet data zone. If an MSID requested in the CDP Configure message was returned as invalid in the CDP Configure Reply message, that MSID will not be included in the CDP. The Local Table is accessed by the CDP to determine the maximum number of sample sizes associated with each MSID.

The data associated with each MSID are delineated in Table 8-14 and graphically represented in Figure 8-13. The information is repeated for each MSID in the CDP in the order of the MSIDs as specified in the CDP Configure message. The time associated with the data in the CDP is contained in the Primary EHS Protocol Header in the Ground Receipt Time field. The Ground Receipt Time is set to the POIC ground time that the CDP was created. If the CDP is being created during a simulation or test, then the POIC ground time will correspond to that simulation or test time.

The CDP contains all the available information associated with the desired MSIDs for the time frame specified for the CDP. Some MSIDs will have multiple samples associated with the time frame as the type of MSID can receive multiple values. The CDP_Configure_Reply message defines the maximum number of samples that can be received within a cycle for that MSID. In Table 8-14, the Overall Samples starts the values for the MSID. To determine the offset between the MSIDs, consult the CDP_Configure_Reply message (section 10.1.1).

Note: The CDP data zone does NOT contain the name of the MSID since that information is available in the CDP_Configure_Reply message.

The Number of Samples in Table 8-14 specifies the current number of samples that is associated with this CDP for this MSID. For example, if “MSIDa” in the CDP_Configure_Reply has a maximum number of 10 samples, yet for this timeframe only four samples were collected, the Number of Samples would be four.

The Status Character for the samples is next in the format of the CDP. In the example, this would mean that you have four status characters. Now, since in the Telemetry Database “MSIDa” is defined as having a maximum of 10 samples, the next 6 (i.e., 10 minus 4) status characters would be zero filled. The priority of the status character returned for a sample is specified in Table 8-17.

Note: When the Number of Samples field is 0, the first Sample Status field will contain a status character corresponding to the Overall Status. The remaining Sample Status fields and all of the Data Sample fields are zero-filled.

Next would come the four values for MSIDa. Once again, since in the Telemetry Database “MSIDa” is defined as having a maximum of 10 samples, the next 6 samples would be zero-filled.

The size of these samples is identified in the CDP_Configure_Reply as the “Sample Size” field and the “Native Sample Size” field. The “Sample Size” field is used to determine the length of the sample size that is returned in the CDP packet. The "Native Sample Size" and the "Sample Size" are returned for every MSID. The Native Sample Size gives the unprocessed MSID length in bits, but this should not be used to obtain the data from the packet because data are on byte boundaries (1 byte increments). The MSID data can be obtained by using the "Sample Size" field that is the size of the data in BYTES. For example, if an MSID is 3 bits long, then the CDP process will place this in a byte. In such a case, "Sample Size" will be 1 byte, and "Native Sample Size" will be 3 bits.

At this point, the information associated with MSIDa is finished for this CDP, and the next MSID would then start with the Overall Status and reiterate the values for that MSID.

The values in the CDPs are based upon the definitions as specified in the Telemetry Database for the different type of parameters. Data in a CDP are inserted on a byte boundary. For example, if the data contain two unprocessed samples of a 3-bit discrete integer, the data would be placed in 2 bytes (with each byte right justified) in the CDP, as specified in Table 8-13.

The CDP is sent to the IP address and port as specified by the remote user. If the remote user requests multiple CDPs, they may all go to the same port address or separate port addresses.

This Page Intentionally Left Blank

9.0 STORED TELEMETRY

The remote user can retrieve stored telemetry in a file. The content of the file can contain individual parameters extracted from a packet by using the short-term storage parameter list request capability (see POIC Capabilities Document (SSP 50304), section 4.1.4.2.1). The content of the file can also contain whole packets or BPDUs (see POIC Capabilities Document (SSP 50304), sections 4.1.4.2.3 and 4.1.4.2.4). This section of the POIC to Generic User IDD describes the stored telemetry file formats for parameters, packets, and BPDUs.

9.1 STANDARD OUTPUT FILE

This section identifies the format of the standard output files used within the POIC. Stored telemetry retrieved using the parameter list request capability are formatted using the standard output file format. Table 9-1 shows the standard output file format. This standard output file format can also be used as the input into a computation or as the output from a computation. Note that to receive the output of the computation in the standard output file format the input of the computation is also required to be in the standard output file format. When the standard output file is created the file will be returned to the workstation that requested the data using whatever connection mechanism was used to connect to the NRT Server (e.g., TCP/IP interface or FTP interface). For the remote user, this means that the standard output file goes to the workstation in the POIC that the remote user is accessing (through X-Windows) to get to the NRT requesting applications or computation. From there, the remote user will need to FTP the standard output file to the remote user's GSE using the FTP capabilities discussed in section 13. Figure 9-1 depicts the process for creating a standard output file.

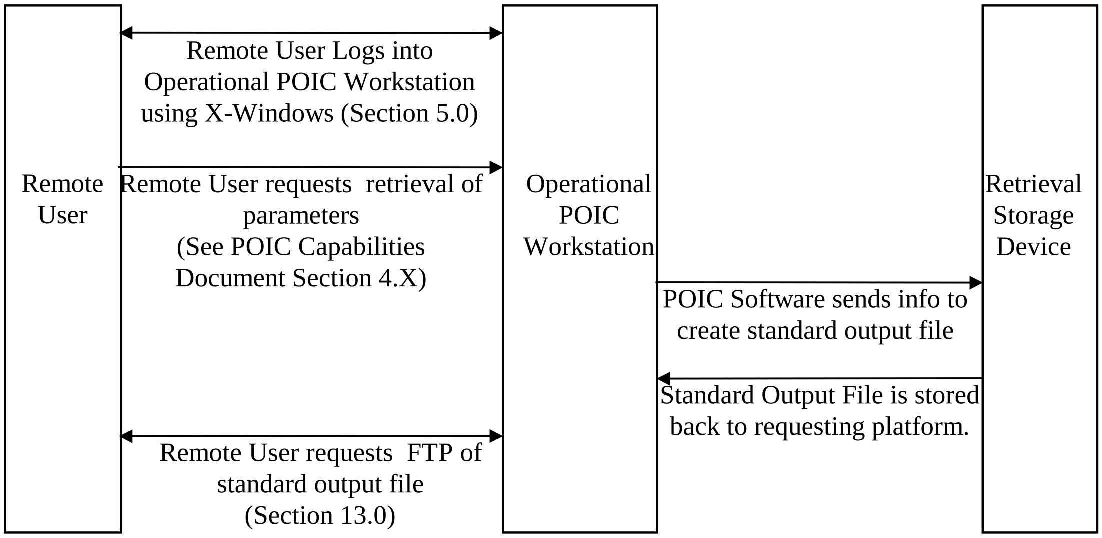

Remote User
Remote User Logs into Operational POIC Workstation using X-Windows (Section 5.0)
Remote User requests retrieval of parameters (See POIC Capabilities Document Section 4.X)
Remote User requests FTP of standard output file (Section 13.0)
Operational POIC Workstation
POIC Software sends info to create standard output file
Standard Output File is stored back to requesting platform.
Retrieval Storage Device

Figure 9-1. Standard Output File Creation

The name of the standard output file will depend upon the name given to the output file by the user. The stored telemetry applications will default the standard output file to the name of the request; however, the user can modify the name of the output file. The extension on the standard output file name is .sto.

a. The output file name is limited within the POIC to 20 characters, with an additional four characters for the extension. The output file name allows alphanumeric characters.

b. The output file format is a <tab> delimited ASCII file.

c. The output file consists of variable length data terminated with either a tab (ASCII Hex Code x09) or new line (ASCII Hex Code x0A) character.

d. Each MSID can have multiple data samples associated with it based upon cycle time. Data Sample 1 through Data Sample n will be repeated for each MSID listed in the file.

e. Each record in the output file corresponds to one EHS data packet.

f. The last data sample for the MSID will be followed by an End_Data marker (#End_Data).

g. File information unique to each file type follows in the file after End Data. This information is multiple lines long.

Table 9-1, Standard Output File Format, shows the standard output file format. The tab control characters being displayed are dependent upon the user's editor. The tab control character shown in Table 9-1 is “>“. The bolded italic text in Table 9-1 represents fixed fields as they will appear in the output file.

The file and request information is included at the end of the file. Table 9-2, Standard Data Interface Data Elements, defines the data elements used within the standard output files.

Table 9-1. Standard Output File Format

#Header	>Timestamp	>MSID 1	>status	> MSID i	>status	>\n
#Data_Type>		>	>Data Type 1	>	>Data Type i	>\n
\n						
#Start_Data \n						
#Data	>Timestamp 1	>data sample 1 of MSID 1	>status 1 of MSID 1	>data sample 1 of MSID i	>status i of MSID I	>\n
	>Timestamp 1	>data sample 2 of MSID 1	>status 2 of MSID 1	>data sample 2 of MSID i	>status 2 of MSIDI	>\n
	>Timestamp 1	>data sample n of MSID 1	>status n of MSID 1	>data sample n of MSID i	>status n of MSID I	>\n
#Data	>Timestamp 2	>data sample 1 of MSID 1	>status 1 of MSID 1	>data sample 1 of MSID i	>status i of MSID I	>\n
	>Timestamp 2	>data sample 2 of MSID 1	>status 2 of MSID 1	>data sample 2 of MSID i	>status 2 of MSID I	>\n
	>Timestamp 2	>data sample n of MSID 1	>status n of MSID 1	>data sample n of MSID i	>status n of MSID I	>\n
#Data	>Timestamp j	>data sample 1 of MSID 1	>status 1 of MSID 1	>data sample 1 of MSID i	>status 1 of MSID I	>\n
	> Timestamp j	>data sample 2 of MSID 1	>status 2 of MSID 1	>data sample 2 of MSID i	>status 2 of MSID I	>\n
	> Timestamp j	>data sample n of MSID 1	>status n of MSID 1	>data sample n of MSID i	>status n of MSID I	>\n
#End_Data \n						
< file unique information>						

Table 9-2, Standard Data Interface Data Elements, contains all the variables specified in the standard output file. The length column gives the maximum length possible for that field. However, the fields can be smaller by just delimiting the field with the tab character. Thus, the values in the fields are left-justified and the field values stop when the tab character is encountered. The columns give the following information about the data:

Data Element: The name associated with the elements in the standard output file.

Description: A description of the field.

Length (bytes): The maximum length of the field

Type: Type is the internal representation of the value. All output is ASCII character; however it is also labeled as decimal if the value is a number.

Format: The format (ASCII, decimal, etc.) associated with the element.

Range: The valid ranges associated with the element

Units: Any Engineering units that may apply to this element.

Res.: Resolution

Table 9-2. Standard Data Interface Data Elements

Maximum Length (bytes)	Field Name	Type	Format	Values	Description
1-255 when present, otherwise 0	Data Sample	depends on value of 'Data Type'	ASCII		Single data sample value of the MSID
4-16	Data Type	character	ASCII	string short unsigned_short long unsigned_long float double time unknown_datatype	string = Character short = Signed Short Integer unsigned_short = Unsigned Short Integer long = Signed Long Integer unsigned_long = Unsigned Long Integer float = Float (single precision) double = Float (double precision) time = Time unknown_datatype = unknown data type
20	MSID	character	ASCII		Telemetry or Pseudo MSID name; i.e., the identifier of a parameter to be retrieved.

Table 9-2. Standard Data Interface Data Elements

Maximum Length (bytes)	Field Name	Type	Format	Values	Description
1 when data is present, otherwise 0	Status	character	ASCII	‘n’, ‘N’ ‘R’ ‘~’ ‘F’ ‘?’ ‘f’ ‘x’ ‘D’ ‘C’ ‘c’ ‘l’ (letter l) ‘d’ ‘Q’ ‘K’ ‘H’ ‘>’ ‘<’ ‘L’ ‘E’ ‘&’ ‘ ’(space) ‘S’	Indicates EHS status of samples: ‘n’ - packet not defined in the routing table ‘N’ - source status - loss of signal (no data returned) ‘R’ - source status - source initialized (data stream replacement - old data returned) ‘~’ - data quality - no data (no returned- old data used) ‘F’ - data quality - dq failed w/ override (new data used - bad data) ‘?’ - data quality - dq suspect w/override (new data used - suspect data) ‘f’ - data quality (dq) - dq failed (no data returned - old data used) ‘x’ - data quality - data is suspect (no data returned - old data used) ‘D’- decommutated (decom) / converted (conv) / calibrated (cal) status - decom error ‘C’- decom/conv/cal status - conversion error ‘c’ - decom/conv/cal status - calibration error ‘l’ - Limit/Expected State (LES) status - LES error ‘d’- Limits not defined ‘Q’ - Format ID error ‘K’ - Parent Frame error ‘H’ - LES status - warning high ‘>’ - LES status caution high ‘<’ - LES status caution low ‘L’ - LES status warning low ‘E’ - LES status - out of expected state ‘&’ - cal sets undefined ‘S’ - old/stale data ‘ ’(space) - source status - acquisition of signal (parameter is ok - new data are returned, where (space) is a single blank character)
17 (GMT) 16 (MET)	Timestamp	integer	ASCII Decimal	If GMT: YYYY:DDD:HH:MM:SS If MET: [-]YY:DDD:HH:MM:SS	Indicates the time period for the associated data depending upon the user requested time type.

9.2 STORED PACKET FILE FORMAT

A stored packet request is used to retrieve whole data packets from a particular protocol/APID for a specified time period. The stored packet is distributed in a file. A more detailed synopsis of this application can be found in the POIC Capabilities Document (SSP 50304) section 4.1.4.2.3. The user must specify the protocol/APID for required data. The user may request the merging of two or more data modes for this type of request. Note that merging is performed on a packet-level basis.

Whole data packets (including all EHS protocol headers) are written to an output file. The bit pattern is compliant with the appropriate EHS protocol defined in section 8 of this document. The file is available through the FTP services discussed in section 13.

The stored packet file format is a file containing multiple packets of any packet type that is encased in the EHS protocol header. This file does NOT contain any additional header information explaining size or content of request. It is only a file containing packet after packet of the same type of data from the specified APID with NO additional header information anywhere as depicted in Table 9-3, Example of Stored Packet File. The end of the packet is determined by parsing the EHS Primary Protocol Header and determining the length of the packet as specified in that header.

The stored packet file contains only the type of data that are requested by the remote user. If the data requested are PDSS Payload CCSDS Packets with a specified APID, then that is the only type of data included. A Ground Ancillary GSE Packet and GSE Data Packet is another CCSDS Packet that can be retrieved in the format by specifying the APID and protocol associated with the GSE Data Packet.

Table 9-3. Example of Stored Packet File

<EHS Primary Protocol Header #1>	<EHS Secondary Header #1>	<CCSDS Packet Headers #1>	<CCSDS Data Zone #1>
<EHS Primary Protocol Header #2>	<EHS Secondary Header #2>	<CCSDS Packet Headers #2>	<CCSDS Data Zone #2>
<EHS Primary Protocol Header n2>	<EHS Secondary Header #n>	<CCSDS Packet Headers #n>	<CCSDS Data Zone #n>

9.3 STORED BPDU FILE FORMAT (TBD9-1)

9.4 PACKET AND BPDU PLAYBACK REQUEST

A packet or BPDU playback request can be performed using the capabilities discussed in the POIC Capabilities Document (SSP 50304) section 4.1.4.2.5. The result of a packet or BPDU playback request is packets or BPDUs in the format as specified in section 8. The playback data is the same as real-time telemetry data, except for the following items:

a. The data mode in the EHS protocol header is specified as one of the many different playback channels.

b. Retrieval Processing Summary Messages (RPSM) are sent instead of UDSMs .

This Page Intentionally Left Blank

10.0 REAL-TIME TELEMETRY REQUESTS

The request mechanism for PDSS payload CCSDS packets and PDSS Bitstream-Protocol Data Unit packets is through the PDL. The initial request mechanism for Ground Ancillary and other GSE packets is through the PDL. Once the user is provided authorization, the user will initiate and terminate GSE packet requests through a Web interface. The initial request mechanism for custom data packets is through the PDL. Once the user is provided authorization, the user will initiate and terminate custom data packets through a programmatic interface. This section describes the CDP request mechanism.

The CDP service provides a programmatic interface for the request and subsequent receipt of user-selected telemetry parameters. The parameters that can be requested through this interface must be defined within the Telemetry database (e.g., telemetry parameters or external pseudo parameters.)

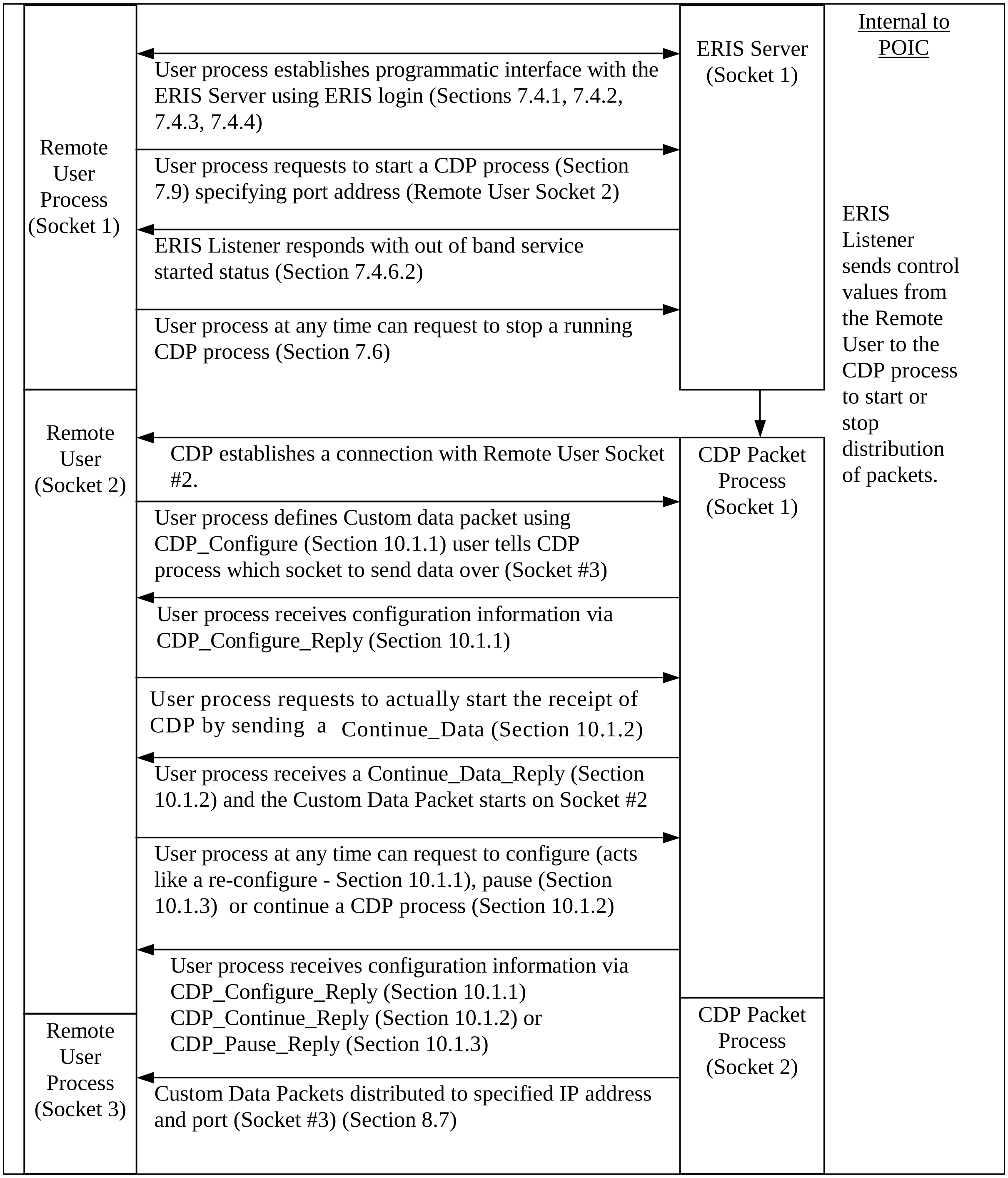
10.1 CUSTOM DATA PACKETS PROGRAMMATIC REQUESTS

Access to and receipt of CDPs is completely provided through a programmatic interface. The interface to the CDP must also be initialized through the programmatic interface identified in section 7, the ERIS interface. The ERIS programmatic interface is initiated with a login process as identified in Figure 10-1, Exchanging CDP Protocols. After the remote user has exchanged the Login information (section 7.4.1) and received the login status (section 7.4.2), the remote user may exchange MOP protocols (section 7.4.3) prior to receiving an ERIS Listener prompt (section 7.4.4). At this time, the remote user can specify to start CDP (section 7.9) and subsequently stop CDP (section 7.6).

NOTICE: In addition to CDPs, the ERIS server also processes all GSE packets and command services (see PGUIDD Volume 2). Together, these data requirements place a sizable load on ERIS resources. The POIC requests that users make efficient use of ERIS resources by expeditiously "stopping" CDP requests when the associated CDP data is not in use.

The CDP process is the server process in a client-server model that provides the capability for a client (remote user process) to request and receive CDPs containing extracted telemetry. The remote user requesting application (client) communicates with CDP (server) through the use of messages transmitted over socket connections utilizing the connection-oriented protocol. CDP transmits the data packets on a second socket connection dedicated solely for this purpose.

There are three requests or messages that the remote user process can use to control the CDP process. The messages include: CDP_Configure, CDP_Pause_Data, and CDP_Continue_Data. For each of the requesting messages, there is a corresponding reply message that is returned to the remote user process to acknowledge the receipt of the request and provide status information on the request. The responses include: CDP_Configure_Reply, CDP_Continue_Data_Reply, and CDP_Pause_Data_Reply.


Remote User Process (Socket 1)
Remote User (Socket 2)
Remote User Process (Socket 3)
ERIS Server (Socket 1)
CDP Packet Process (Socket 1)
CDP Packet Process (Socket 2)
Internal to POIC

User process establishes programmatic interface with the ERIS Server using ERIS login (Sections 7.4.1, 7.4.2, 7.4.3, 7.4.4)

User process requests to start a CDP process (Section 7.9) specifying port address (Remote User Socket 2)

ERIS Listener responds with out of band service started status (Section 7.4.6.2)

User process at any time can request to stop a running CDP process (Section 7.6)

ERIS Listener sends control values from the Remote User to the CDP process to start or stop distribution of packets.

CDP establishes a connection with Remote User Socket #2.

User process defines Custom data packet using CDP_Configure (Section 10.1.1) user tells CDP process which socket to send data over (Socket #3)

User process receives configuration information via CDP_Configure_Reply (Section 10.1.1)

User process requests to actually start the receipt of CDP by sending a Continue_Data (Section 10.1.2)

User process receives a Continue_Data_Reply (Section 10.1.2) and the Custom Data Packet starts on Socket #2

User process at any time can request to configure (acts like a re-configure - Section 10.1.1), pause (Section 10.1.3) or continue a CDP process (Section 10.1.2)

User process receives configuration information via CDP_Configure_Reply (Section 10.1.1) CDP_Continue_Reply (Section 10.1.2) or CDP_Pause_Reply (Section 10.1.3)

Custom Data Packets distributed to specified IP address and port (Socket #3) (Section 8.7)

Figure 10-1. Exchanging CDP Protocols

If a socket connection is lost after the socket connection has been established and the CDPs are being received by the remote user, then the CDP process will attempt to reestablish the socket.

a. If the Command socket is dropped, then CDP process will perform the following:

    (1) If sending data packets, then stop.

    (2) Try to reestablish command socket connection. If successful, then place CDP in the Initialized state. If connection fails, then exit the CDP process with appropriate cleanup.

b. If Data socket is dropped, then perform the following:

    (1) Stop sending packets.

    (2) Place CDP in Initialized state awaiting "CDP_Configure" request.

If the CDP has received a reconfiguration signal notifying the POIC process that some reconfiguration has occurred within the POIC, then the following items might occur:

a. If the reconfiguration signal specified that a database in use has been changed, then the CDP process stops sending data packets and waits for a CDP_Configure Request.

b. If the reconfiguration signal specifies that any of the MOP has changed, then the Custom Data Packet process is terminated.

c. If any other reconfiguration signal has been received, then the CDP processing continues without affecting CDP.

10.1.1 CDP CONFIGURE

The CDP request is initiated by the remote user process by sending a CDP_Configure message to the CDP process. The message contains the information needed by CDP to establish the data socket connection and to prepare packets for transmission. The CDP uses a TCP/IP client server connection oriented protocol for the transfer of messages to and from the CDP server process. Users are required to provide the message structure defined in Table 10-3, CDP Configure, for this request. CDP messages are right/left-justified as shown in Table 10-1.

Upon receipt of the CDP_Configure message from the remote user process, the CDP process performs the necessary processing to configure itself for data packet distribution. The CDP process then returns a CDP_Configure_Reply message to the remote user process acknowledging receipt of the CDP_Configure message. The CDP_Configure_Reply message contains information needed by the remote user process to determine the success of the configure request and also how to extract the data from the CDP packets. Users will be provided the message structure defined in Table 10-4, CDP Configure Reply.

If the CDP process receives a subsequent CDP_Configure message from the remote user process, then delivery of the CDP is stopped until the CDP process is reconfigured with the new CDP format. A CDP_Configure_Reply is sent to the remote user process and the new format of the CDP is restarted on the specified port address.

Table 10-1. Data Type Justification for CDP messages

Data Type	Size in Bits	Justification	Fill value (hex)
unsigned short	16	Right	zeros
unsigned int	32	Right	zeros
unsigned char	8	Right	zeros
Null Terminated ASCII String	8 per character	Left	NULL (x0)

The CDP_Configure message starts with the command identifying it to the CDP process as the CDP_Configure message. The Data Address field identifies the IP address of the remote user GSE that will be receiving the CDP. The IP address will hold a 32-bit address or a 16-bit address with a fill pattern of zeros on the left.

The Port Number field identifies the requested port number from the remote user that will be waiting to receive the CDP. As identified in Figure 10-1, the data are transferred on a different port than the CDP commands. This allows the control and data to be transferred at the same time. The port number can also be either a 32-bit port or a 16-bit port. If the port field is only 16 bits, the 16 MSB are zero filled.

The update rate specifies how much time will occur between the instance of creation and distribution between two packets of CDPs. The remote user can request that the CDP be transmitted for a limited duration or for an unlimited amount of time. To specify a limited duration, the Duration field is filled with the appropriate timeframe. This is a relative time from the reception of the CDP_Continue request.

The Data Mode field contains the requested source of data to fill the CDP. If the Data Mode is REALTIME, then the data that are being processed as real-time data in the POIC will be used to generate the CDP. The Dump and Playback Modes are for the data that are processed from either a LOR dump, a COR dump, a short-term playback, or long-term playback of data. The Data Mode field correlates to the Data Mode field in the Primary EHS Protocol Header (see section 8.1). If the field value is less than 10 characters, then the field is left-justified and filled with NULL(\0) characters.

The database field identifies from which database the MSIDs are to be decommutated. The databases are specified by either BASELINE or ARCHIVE. In order for the remote user to determine which version of the database corresponds with these values, the remote user either requests the Common Configuration programmatic interface (see section 7.7.1) or uses the Database Operational Support History application to visually determine which is desired.

The Num MSIDs field identifies how many MSIDs are requested by the remote user for each CDP. The number also is used in the CDP_Configure message to parse out how many sets of the remaining values. For example, if you set the Num MSIDs to 5, then you would have 5 sets of “MSID Name, First or All Samples and Functions.” The maximum number of MSIDs is based upon the following variables:

a. Number of samples associated with the MSIDs (this corresponds to how often the values are retrieved per second).

b. Size of the sample in bytes.

The limitation for the number of MSIDs in a CDP is the limit on the actual packet size based upon these considerations. The equation is as identified in Equation 10-1.

L = the limitation on the Packet data zone size (see section 8.7).

N Maximum Samples = number of samples associated with the MSID per second.

Sample Size = size of the sample in bytes based upon processing type (See Table 10-2).

M= the number of MSIDs.

The constant "8" in Equation 10-1 represents four bytes for the Overall Status field and four bytes for the Number of Samples field.

$$L = \sum_{M=1}^{M = \#\text{ of MSIDs}} 8 + (\text{N Maximum Samples} * \ (1 + \text{Sample Size})\ )$$

Equation 10-1. Limitation on Number of MSIDs for each CDP

The Sample size of the MSID depends upon how the data are processed. Table 10-2 shows how to calculate sample size depending on whether the data is unprocessed (raw), converted to a standard format, or calibrated. The data processing method is set by the user in the "Function" Field in the CDP _Configure request message as shown in Table 10-3.

Table 10-2. Sample Size Table Calculations

	Unprocessed Data	**Data Converted To**	**Data Calibrated To**
String	See MSFC-STD-1274B Vol. 2 Appendix B. (Data code starts with S) (SASC SASCB SEBC	SASC - Null term ASCII String that is 8 bits per character	Not Applicable.
	SUND	Not applicable	Not applicable
Integers	See MSFC-STD-1274B Vol. 2 Appendix B. (Data code starts with I) (IMAG ITWO ITWOB	ITWO - Two's Complement Integer that is either 16 or 32 bits.. If unprocessed length is less than or equal to 16 bits, converted value is 16 bits; else converted	FEEE - IEEE Floating Point (double precision-64 bits)

Table 10-2. Sample Size Table Calculations

	Unprocessed Data	Data Converted To	Data Calibrated To
	ITWOW ITWOX IDSI)	value is 32 bits.	
	IDIS	IUNS - Unsigned Integer that is either 16 or 32 bits. If unprocessed length is less than or equal to 16, converted length is 16; else converted length is 32.	SASC for state code conversion (13 bytes)
	IUNS, IBCD IUNSB, IUNSW IUNSX	IUNS - Unsigned Integer that is either 16 or 32 bits. If unprocessed length is less than or equal to 16 bits, converted value is 16 bits; else converted value is 32 bits.	FEEE - IEEE Floating Point (double precision – 64 bits)
	IUND	Not applicable	Not applicable
Floats	See MSFC-STD-1274B Vol. 2 Appendix B. (Data code starts with F) (FEEE FIBM FMIL FNTL FSPL FVAX)	FEEE - IEEE Floating Point (single precision – 32 bits, double precision – 64 bits) If unprocessed length is less than or equal to 32 bits, converted value is 32 bits; else converted value is 64 bits.	FEEE - IEEE Floating Point (double precision – 64 bits).
Times	See MSFC-STD-1274B Vol. 2 Appendix B (Data code starts with T) (TDMS TECI TECS TEHS TERT TGMT TGPC TIUS TOOI TTSM TTWO TISS TUDS)	TECT - EHS Converted Time, which is 40 bytes	Not applicable

The MSID name is specified in the POIC Telemetry Database as a 20-character element. For this field value, the 20 characters are followed by a <NULL> terminator. If the remote user has specified a MSID name that is less than 20 characters, the field value will be left-justified and filled in with NULL (\0) characters. For the valid characters for an MSID name, see MSFC HOSC Database Definitions, Volume 1, Telemetry Databases, MSFC-DOC-1949, Appendix C.

The remote user should be familiar with the MSIDs available that can be requested by the remote user. If not sure of the MSID name, the remote user can perform one of the following:

a. Access the Web application Telemetry Database and perform a query on the MSIDs that the user is authorized to access (see POIC Capabilities Document (SSP 50304), Section 4.4.1). The remote user can see which MSIDs are available and use the method to create the CDP_Configure information.

b. Request that the Web application Telemetry Database create a database ASCII file of the database values and download the file onto the remote user's GSE to be read into the remote user's application. The file may be very large. The user may want to only create a partial Telemetry Database file so that it is limited to the values that the remote user is interested in. (See POIC Capabilities Document (SSP 50304) Section 4.4.1.)

The First or All Samples field is used by the remote user to specify if all samples associated with the MSID should be returned in the CDP or just the first sample received during the user specified update rate/cycle. If all samples are specified in the CDP_Configure message, then the maximum number of samples for each MSID is returned in the CDP_Configure_Reply message.

The Function field allows the remote user to specify what sort of processing is performed on the requested MSID. The options are as listed:

a. Unprocessed

b. Converted

c. Calibrated

d. Limit/Expected State (LES)

The remote user can request to have a combination of Converted and LES or Calibrated and LES performed on the MSID. The remote user specifies the value by the following:

00 00 00 01=unprocessed w/o LES Status
00 00 00 02=converted w/o LES Status
80 00 00 02=converted w LES Status
00 00 00 04=calibrated w/o LES status
80 00 00 04=calibrated w/ LES status

If the data in the CDP are converted or calibrated, then the data type of the output data is dependent upon what format is specified for the MSID in the POIC telemetry Database.

The end of the CDP_Configure request is calculated based upon the fixed number of bytes at the beginning of the message and then the NumMSID value used in the middle of the request. There is a standard number of bytes associated with each MSID. The end of the CDP_Configure request is reached when the CDP process has read the number of bytes in Equation 10-2.

Length of CDP_Configure request = 43 bytes + (Num MSIDs * 26 bytes)

Equation 10-2. CDP_Configure request Length

So be sure to fill in all 26 bytes for each specified MSID according to the Num MSIDs in the CDP_Configure request. For example, your MSID value only contains 13 characters instead of 20. The CDP_Configure Request will read up to the first null terminator in the 21 characters (20 characters of legal entry + 1 character for the null terminator) and then put that value as the MSID name. Then the CDP_Configure Request will “skip” up until byte #22 and read that value for the “First or All Samples” value. The next 4 bytes are read in to determine the processing performed on that MSID. Then the process starts again with the next MSID on the list.

Table 10-3. CDP Configure

Byte	Bit	Field Name	Type	Values	Description
0-1	15-0	Command	unsigned short	1 (decimal)	CDP_CONFIGURE = 1
2-5	31-0	Data Address	unsigned int	> 0 (decimal)	Address Of Remote User Process Note: Value Must Have Been Pre-Approved In The POIC Account Request Form.
6-9	31-0	Port Number	unsigned int	> 0 (decimal)	Port of Remote User Process Note: Value must have been pre-approved in the POIC account request form.
10-13	31-0	Update Rate	unsigned int	>= 1 (seconds between cycles) (decimal)	time between pkt output
14-15	15-0	Duration to run-days	unsigned short	0 – 365 (decimal) Bits 15-10 are always set to zero.	To specify infinite duration the value of each of these fields must be 0
16	7-0	Duration to run-hour	unsigned char	0 – 23 (decimal) Bits 7-5 are always set to zero	To specify infinite duration the value of each of these fields must be 0
17	7-0	Duration to run-minute	unsigned char	0 – 59 (decimal) Bits 7-6 are always set to zero.	To specify infinite duration the value of each of these fields must be 0
18	7-0	Duration to run-second	unsigned char	0 – 59 (decimal) Bits 7-6 are always set to zero	To specify infinite duration the value of each of these fields must be 0
19-29	87-0	Data Mode	Null Terminated ASCII String	“REALTIME ” “DUMP1” “DUMP2” “DUMP3” “PLAYBACK1” “PLAYBACK2” “PLAYBACK3” “PLAYBACK4” “PLAYBACK5” “PLAYBACK6” “PLAYBACK7” “PLAYBACK8” “PLAYBACK9” “PLAYBACK10” “PLAYBACK11”	where to collect data from

Table 10-3. CDP Configure

Byte	Bit	Field Name	Type	Values	Description
30-38	71-0	Database	Null Terminated ASCII String	"ARCHIVE" "BASELINE"	
39-42	31-0	Num MSIDs	unsigned int	1 - Equation 10-1(decimal)	number of MSIDs being sent in the block
43-63	167-0	MSID name	Null Terminated ASCII String	Valid MSID from specified POIC TDB	Name of the MSID
64	7-0	First or All Samples	unsigned char	0 = Get the first sample 8 = Get all the samples (decimal)	
65-68	31-0	Functions	unsigned int	00 00 00 01 (hex) =unprocessed w/o LES Status 00 00 00 02 (hex) =converted w/o LES Status 80 00 00 02 (hex) =converted w LES Status 00 00 00 04 (hex) =calibrated w/o LES status 80 00 00 04 (hex) =calibrated w/ LES status	Valid combinations of functions: Convert or LES Calibrated or LES

Note: MSID name, num-of-sample, and functions are recurring. Example: If you set the num_msids to five, then you would have five sets of MSID name, num-of-sample, and functions.

Note: The remote user will receive a CDP_Configure_Reply in response to the CDP_Configure message on the same socket/port connection that the CDP_Configure message was sent. The Command field on the CDP_Configure_Reply verifies that the reply is from the CDP_Configure message. The Configure_Status field is a general status field for the CDP_Configure request as a whole. If anything besides an OK is received, then the CDP_Configure request failed because of the specified Configure_Status and the remainder of the CDP_Configure_Reply message will be NULL (blank) filled. However, if the Configure_Status is No Enabled MSIDs, then the CDP_Configure_Reply message will contain the name and status of each failed MSID in the Configure Request.

- Invalid Command

Check the type of the value sent. This variable should be an "unsigned short"

- Invalid Data Address

Data Address given in the CDP_Configure is not a valid value. Check to ensure that the data address is of the right format (e.g., dotted decimal) and non-negative.

- Invalid Data Port

Data Port given in the CDP_Configure is not a valid value. The number must be non-negative.

- Invalid Packet Rate

Check the type of the value sent. The variable should be an “unsigned int.” The value should be greater than 1 without a decimal point or anything else.

- Invalid Time

Check the format of the value sent. Remember that all values have to be zero to specify that the time is unlimited. See Table 10-3, CDP Configure, for details on the configuration of the time value.

- Invalid Data Mode

Check the format of the Data Mode value sent on the CDP_Configure request. Make sure that the case and spacing is followed exactly. Make sure it is followed by a <NULL> value.

- Invalid Database

Check the format of the Database value sent on the CDP_Configure request. Make sure that the case and spacing is followed exactly. Make sure it is followed by a <NULL> value.

- Invalid Number MSIDs

NumMSIDs given in the CDP_Configure is not a valid value. The number must be non-negative.

- Packet size too Large

The calculations based upon the number of MSIDs, maximum sample size for each MSID, and MSID data type created a packet larger than the maximum packet size allowed. Reduce the number of MSIDs in the request or reduce the number of samples requested according to cycle.

- Maximum Bandwidth Exceeded

The calculations based upon the number of MSIDs, maximum sample size for each MSID, and MSID data type created a packet larger than the allowable bandwidth for the specified IP address. Reduce the number of MSIDs in the request or reduce the number of samples requested for each cycle.

- Memory Allocation Error

The error is received when an internal buffer that is used to decommutate the MSIDs failed to be allocated properly for the MSID. Either resubmit the CDP_Configure request or call the POIC Help Desk and inform them about receiving this error message because the ERIS Server has run out of room to process CDP requests.

- No Enabled MSIDs

The MSIDs status in the CDP_Configure_Reply is checked, and if all the MSIDs status contained an error then there are no enabled or valid MSIDs to be returned in the CDP. When the No Enabled MSIDs status is returned, the CDP_Configure_Reply will also contain the individual MSID statuses as described in Table 10-4.

- Data Connection Error

CDP process tried to connect to the specified data port and received a connection error. Check the value on the Data Port value sent in the CDP_Configure and confirm that it is the right value.

- Configure Reply Too Large

The size of the Configure_Reply message will be too large to be passed back to the remote user process. Reduce the number of MSIDs requested and try again.

The APID value returned is the APID that will be used in the CCSDS Primary Header for CDPs. The APID will identify the CDP request over a potential other CDP request going to the same IP address over the same port. The APID is dynamically generated at the POIC and is not guaranteed to be unique from the ISS generated APIDs. Neither is it guaranteed to match any APIDs that were assigned as part of the ISS PDL process. The CDP APID is generated on a “next available” number basis and can be different every time a CDP is requested. For example, if a CDP is requested and then the remote user terminates that request (i.e., sends a STOP command), then reinitiates the same CDP request, the APID may differ because it is considered a different CDP request.

The remainder of the information on the CDP_Configure_Reply is information based upon the MSIDs that were requested in the CDP_Configure message. The order of the MSIDs in the CDP_Configure_Reply is exactly the order specified in the CDP_Configure. The MSID is listed and all information is given concerning that specific MSID. The information is then repeated for the next MSID in the list. The CDP Configure Reply message is completed when the remote user process has encountered the number of bytes as specified in Equation 10-3.

Length = 6 bytes + (34 bytes * Num MSIDs from CDP_Configure)

Equation 10-3. CDP_Configure_Reply Length

The MSID field contains the MSID value as specified in the CDP_Configure request. The field is in the same format as specified in the CDP_Configure section. The field is <NULL> terminated.

The Status field corresponds to the status of the individual MSID requested in the CDP_Configure request. If the value of the status is OK, then the information about the MSID follows the status message. If the value of the status field is anything other than OK, then the next 12 bytes will be not applicable. They will be NULL (blank) filled and can be ignored. If the value of the status field is anything other than OK, the MSID will NOT be included in the CDP when it is started.

The MSID Offset field identifies the offset for the MSID in the CDP data zone from the start of the data zone packet. This does not include any of the headers (e.g., CCSDS headers or EHS Protocol headers). It is assumed that the first byte in the CDP data zone is considered byte 0 and the offset number can be added to the starting byte of the CDP data zone to get directly to the MSID information within the CDP.

The EHS data type field will give a numeric value that can be converted to the assigned EHS data type according to the information provided in Table 10-4, CDP Configure Reply. The EHS data type field corresponds with the MSID if the "Function" field on the requested MSID in the CDP_Configure message was specified as being "Converted" or "Calibrated". If the MSID "Function" field in the CDP_Configure message is "Unprocessed", then the EHS data type field will be set to 0.

The Native data type is the data type associated with the MSID in the POIC Telemetry Database as the unprocessed format when it comes down in the telemetry stream. If the MSID "Function" field specified to receive the MSID as "Unprocessed", this will be the format for the MSID that is in the CDP. The size and more information associated with each of these Native Data Types can be found in MSFC-STD-1274B, Volume 2, Appendix B.

There are various time structures from the Native data type structures that can be valid time values. These are specified in MSFC-STD-1274, Appendix B as time types. If these time types are converted or calibrated they are changed to the EHS Converted Time data type as defined in MSFC-STD-1274 Appendix B, for processing within the EHS.

The Max Num Samples field specifies the maximum number of samples that can be gathered for the MSID. The value is used in the CDP format to determine the maximum number of samples that will be identified with each MSID. The Sample Size is the number of bytes that each sample will take up within the CDP when retrieving samples for that MSID. The Sample Size is dependent upon the type of functions (processing) requested when asking for the MSID. The Native Sample Size is the size in bits of the sample when it is unprocessed (i.e., unprocessed telemetry). This is the size of the sample if the MSID value is being retrieved in an unprocessed state.

Table 10-4. CDP Configure Reply

Byte	Length	Field Name	Type	Values	Description
0-1	2	Command	unsigned short	1	CDP_Configure
2-3	2	Configure Status	unsigned short	0=OK 1=Invalid Command 2=Invalid Data Address 3=Invalid Data Port 4=Invalid Packet Rate 5=Invalid Time 6=Invalid Data Mode 7=Invalid Database 8= Invalid Number MSIDs 9=Packet Size Too Large 10=Maximum Bandwidth Exceeded 11=Memory Allocation 12= No Enabled MSIDs 13= Data Connection Error 14= Configure Reply Too Large (all values decimal)	
4-5	2	APID	unsigned short	0 – 2047 (decimal)	Unique packet identifier
6-26	21	MSID Name[1]	Null Terminated ASCII String		Name of MSID

Table 10-4. CDP Configure Reply

Byte	Length	Field Name	Type	Values	Description
27	1	Status[1]	char	'\0' = OK 'G' Packet Routing Table Configuration Error 'I' (UITM Process Connect Error) 'B' Common Configuration Error 'W' Memory Error 'A' Distribute Packets (DP) Process Not Responding 'a' Context Dependent Decom (CDD) Process Not Responding 'T' Telemetry Database Discrepancies '&' Calibrated Sets not defined in Local Table 'P' Telemetry Processing Discrepancies 'g' (Group Activation Manager (GAM) Process not responding) 'Z' Unrecognized Status From TNS_Initialize_ Decom	
28-31	4	MSID Offset[1]	unsigned int	4 – Equation 10-1 (decimal)	Location in packet. Offset does NOT include header. This is in bytes
32	1	EHS Data Type[1]	unsigned char	0 – 9 (decimal) Where: 0 = None 1= 8 bit ASCII string 2= IEEE double precision (64 bits) 3 = EHS Converted Time (320 bits – 40 bytes) 4 = IEEE single precision (32 bits) 5 = 32 bit two's-complement integer 6 = 32 bit unsigned integer 7 = 16 bit two's-complement integer 8 = 16 bit unsigned integer 9 = Unknown	The data type used by the EHS software for Converted and Calibrated data as output data types associated with the MSID.

Table 10-4. CDP Configure Reply

Byte	Length	Field Name	Type	Values	Description
33	1	Native Data Type[1]	unsigned char	0-37 and 255 (decimal) Where: 0=FEEE 1=FIBM 2=FMIL 3=FNTL 4=FSPL 5=FVAX 6=IDIS 7=IMAG 8=ITWO 9=ITWOW 10=IUND 11=IUNS 12=IDSI 13=IBCD 14=SASC 15=SASCB 16=SEBC 17=SUND 18=TDMS 19=TECI 20=TECS 21=TEHS 22=TERT 23=TGMT 24=TGPC 25=TIUS 26=TOOI 27=TTSM 28=TTWO 29=TECT 30=TISS 31=TUDS 33=IUNSB 34=IUNSW 35=IUNSX 36=ITWOB 37=ITWOX 255=Unknown	The MSIDs native data type (prior to conversion to EHS data type) Refer to MSFC-STD-1274 for more information on data type definitions
34-35	2	Max Num Samples[1]	unsigned short	1 - 65535 (decimal)	The maximum number of samples for the MSID
36-37	2	Sample Size[1]	unsigned short	1 - 65535 (decimal)	The size in bytes of each sample based on processing
38-39	2	Native Sample Size[1]	unsigned short	1 - 65535 (decimal)	The size in bits of the sample (independent of processing)

[1] This set of entries will occur for each MSID. All information is provided on an MSID by MSID basis.

10.1.2 CDP CONTINUE DATA

The CDP_Continue_Data message is sent to the CDP process by the requesting process once a successful configuration has been completed and acknowledged by CDP. The message is used to signal the CDP process to begin the transmission of data packets across the data socket connection. The CDP_Continue_Data message contains no information other than a message type indicator as detailed in Table 10-5, CDP Continue Data. The CDP_Continue_Data message is 2 bytes long.

The remote user process must have an ERIS interface established with the POIC as described in Section 7 to request a CDP Continue Data. The CDP uses a TCP/IP client-server connection oriented protocol for the transfer of messages to and from the CDP server process (See Figure 10-1, Exchanging CDP Protocols, for the appropriate timeframe for the remote user process to request a CDP Continue Data Request). Users are required to provide the message structure defined in Table 10-5, CDP Continue Data, for the request.

Table 10-5. CDP Continue Data

Byte	Bit	Field Name	Type	Values	Description
0-1	15-0	Command	unsigned short	2 (decimal)	CDP_Continue

The CDP_Continue_Data_Reply message is returned to the remote user process by CDP as an acknowledgment of the receipt of the CDP_Continue_Data message. The message contains the status of the CDP_Continue_Data. The CDP_Continue_Data_Reply message is 4 bytes long. Users will be provided the message structure defined in Table 10-6, CDP Continue Data Reply.

The Status field corresponds to the status of the CDP_Continue Request. If the value of the status is OK, then the CDPs will begin flowing on the specified port from the CDP_Configure request. If the value of the status field is anything other than OK, then the CDP is not being sent for one of the following reasons:

a. Invalid Command

   This command received by the CDP Process is not a valid command at this point and time.

b. Not Configured

   The CDP Process has not received a valid CDP_Configure Request to initiate the CDP process, or the CDP_Configure Request that was received was not processed properly or had a serious error. Check the status of the CDP_Configure_Reply status to determine if a serious error occurred on the CDP_Configure request.

Table 10-6. CDP Continue Data Reply

Byte	Bit	Field Name	Type	Values	Description
0-1	15-0	Command	unsigned short	2 (decimal)	CDP_Continue
2-3	15-0	Status	unsigned short	0=OK 1= Invalid Command 2= Not Configured (all values decimal)	

10.1.3 CDP PAUSE DATA

The CDP_Pause_Data message is sent to the CDP process by the requesting process to pause the transmission of data packets. Like the CDP_Continue_Data message, the CDP_Pause_Data message contains no information. Receipt of this message by the CDP process causes CDP to pause the transmission of data packets on the data socket connection. The CDP_PAUSE_DATA message is 2 bytes long.

The remote user must have an ERIS interface established with the POIC as described in Section 7 to request a CDP Pause Data. The CDP uses a TCP/IP client-server connection oriented protocol for the transfer of messages to and from the CDP server process (See Figure 10-1, Exchanging CDP Protocols, for the appropriate timeframe for the remote user process to request a CDP Pause Data Request). Users are required to provide the message structure defined in Table 10-7, CDP Pause Data, for the request.

Table 10-7. CDP Pause Data

Byte	Bit	Field Name	Type	Values	Description
0-1	15-0	Command	unsigned short	3 (decimal)	CDP_Pause

The CDP_Pause_Data_Reply message is returned to the remote user process by CDP upon receipt of a CDP_Pause_Data message and contains information on the status of the request to pause data packet transmission. The CDP_Pause_Data_Reply message is 4 bytes long. Users will be provided the message structure defined in Table 10-8, CDP Pause Data Reply.

The Status field corresponds to the status of the CDP_Pause Request. If the value of the status is OK, then the CDPs will stop flowing on the specified port from the CDP_Configure request. If the value of the status field is anything other than OK, then the CDP is not being stopped for one of the following reasons:

a. Invalid Command

   The command received by the CDP Process is not a valid command at this point and time.

b. Not Running

   The CDP Process is not currently distributing CDPs; therefore, the distribution of the CDPs cannot be stopped.

Table 10-8. CDP Pause Data Reply

Byte	Bit	Field Name	Type	Values	Description
0-1	15-0	Command	unsigned short	3 (decimal)	CDP_Pause
2-3	15-0	Status	unsigned short	0 = OK 1= Invalid Command 2= Not Running (all values decimal)	

11.0 COMMAND STATUS INTERFACES

The commanding capabilities provided at the POIC are defined in the POIC Capabilities Document (SSP 50304), Section 4. The programmatic capabilities to update and uplink a command are considered a secure service and are defined in SSP 50305, Volume 2, POIC to Generic User Interface Definition Document: Secured Services.

This Page Intentionally Left Blank

12.0 DATABASE FILE FORMATS

This section addresses the file formats for partial database deliveries of POIC command and telemetry database information to remote users and for the Ground Support Equipment (GSE) Packet Definition File.

12.1 PARTIAL DATABASE FILE FORMAT

The POIC provides a partial database download capability for the remote user who requires POIC telemetry and/or command database information at their local site. For the telemetry database, the remote user can request partial database information for an entire packet based on an APID, or can narrow the request to individual parameters (MSIDs) within a packet (TBR 12-1). Partial database information is delivered to a user in an ASCII text file. The ASCII text file format is described in MSFC-DOC-1949 Volume 4 section 3.2.1. The ASCII text file is delivered to the remote user's local site via FTP. The list of delivered ASCII file names for a partial telemetry database download is provided below, along with the MSFC-STD-1949 Volume 4 database table associated with each file. Note that the "nnnn" prefix in each file name is a sequence number generated by the database to ensure file uniqueness. The file names for a partial command database download are TBR 12-2.

Partial Telemetry Database ASCII File Name	**MSFC-STD-1949 Table**
nnnn_cal_switch.txt	*Calibration Switch Table*
nnnn_counter.txt	*Counter Table*
nnnn_es_switch.txt	*Expected State Switch Table*
nnnn_exp_state.txt	*Expected State Table*
nnnn_lim_switch.txt	*Limit Switch Table*
nnnn_limit.txt	*Limit Table*
nnnn_msid.txt	*Measurement Table*
nnnn_msid_loc.txt	*MSID Location Table*
nnnn_msid_samp.txt	*MSID Sampling Table*
nnnn_owner.txt	*Owner Table*
nnnn_packet.txt	*Packet Table*
nnnn_packet_fmt.txt	*Packet Format Table*
nnnn_point_pair.txt	*Point Pair Table*
nnnn_poly_cal.txt	*Polynomial Calibration Table*
nnnn_state_code.txt	*State Code Table*
nnnn_stream_id.txt	*Stream ID Table*
nnnn_subset.txt	*Subset Table*
nnnn_subset_fmt.txt	*Subset Format Table*
nnnn_tlm_sys.txt	*Telemetry System Table*

.

12.2 DATABASE CHANGE REQUEST ATTACHMENT FORMAT

The POIC Capabilities Document (SSP 50304) in section 4.4.5 explains the Database Change Request capability in the POIC. The following is the format for providing database changes to existing command and telemetry database content in the POIC:

- ASCII file formats of the databases as defined in MSFC-DOC-1949

12.3 GSE PACKET DEFINITION FILE FORMAT

The GSE Packet Definition File is provided to support a remote user who will receive GSE packets (see section 8.6) from the POIC. The GSE Packet Definition file contains the following information for each GSE packet stored in the POIC Telemetry Database:

a. Packet ID and description

b. Packet length

c. GSE Packet format

d. POIC Telemetry Database which the GSE has been validated against

e. MSID names and locations within the GSE packet

f. MSID length

g. MSID sampling information

h. MSID processing information

Note that the GSE Packet Definition File only contains information related to the GSE Packet data zone. EHS and CCSDS headers are as defined in section 8. The GSE Packet Definition File naming convention is "*packetid#*_packet_*formatid#*_format.txt".

The GSE Packet Definition File format is defined in the following tables with the following rules for formatting the ASCII text files.

a. A separate file is created for each GSE Packet format.

b. The data for each row from the GSE Packet Table is terminated by an end-of-line symbol (\n = ASCII value 10).

c. Each data value is delimited by a comma (, = ASCII value 42)

d. *White space* is defined as one or more spaces, or horizontal tab characters. White space between data values is ignored.

e. Blank lines between data values are ignored.

The GSE Packet Definition File consists of the information depicted in Figure 12-1. The information contained in the GSE Packet Information is depicted in Table 12-1. The information for each MSID is depicted in Table 12-2 and is repeated for each MSID in the Packet. The order of the MSIDs in this file is the order of the MSIDs that will be distributed in the GSE Packet (see section 8.6). The GSE Packet Information is provided for a single format of the GSE Packet.

GSE Packet Information	MSID Information #1	MSID Information #2 …..	MSID Information #N

Figure 12-1. GSE File Format

The width column in Table 12-1 gives the maximum width, in bytes, allowed for that element. However, since each of these elements is controlled by the comma (,) delimiter, the file will contain the actual value followed by a comma with no fill of blanks before the delimiter. There is no white space after the delimiter.

Table 12-1. GSE Packet Information

Width (Bytes)	Field Name	Type	Values	Description
4	GSE Packet ID	Integer	0-2047	GSE Packet ID will be based on predefined list of available APID (s) that are assigned by Ground Support Requirements Team (GSRT) for each user.
1	Format ID	Integer	0-7	Format ID can be used to indicate a different version of the same packet.
6	Total Packet Length	Integer	39-61,438	Total length will be calculated based on the MSIDs defined and the additional status fields that are provided by EHS. Total Packet Length includes all headers (EHS and CCSDS)
2	Output Rate	Integer	1-60	Output rate indicates how often the packet will be transmitted from the EHS system. This value is defined as units in seconds.
250	Description	Char	See MSFC-DOC-1949, Volume 4, Appendix B	Description of the GSE Packet
4	Valid Database Version #1	Char	Alphanumeric set of A.Z (uppercase characters), and 0.9.	Revision of the Telemetry Database that this GSE Packet is valid against (First of three)

Table 12-1. GSE Packet Information

Width (Bytes)	Field Name	Type	Values	Description
4	Valid Database Version #2	Char	Alphanumeric set of A.Z (uppercase characters), and 0..9.	(Second of three)
4	Valid Database Version #3	Char	Alphanumeric set of A.Z (uppercase characters), and 0..9.	(Third of three)

For each MSID in the GSE Packet, the information in Table 12-2 is provided to the remote user. The width column, in bytes, gives the maximum width allowed for that element. However, since each of these elements is controlled by the comma (,) delimiter, the file will contain the actual value followed by a comma with no fill of blanks before the delimiter. No white space is provided after the delimiter.

Table 12-2. MSID Information

Width (Bytes)	Field Name	Type	Values	Description
20	MSID	Char		MSID in the GSE Packet
6	MSID Start Octet	Integer	40-61437	MSID Start Octet will be calculated from the beginning of the Primary EHS Protocol Header and will point to first byte of the MSID Overall Status field in the packet.

1	Sample Flag	Char	A= All Samples
			F= First Sample
				Specifies whether only one sample will be sent of the MSID or all of the samples that are available for this MSID.

2	Processing Flag	Char	CA = Calibrated
			CO = Converted
			UN = Unprocessed
				Specifies the type of processing that is applied to all of the samples of this MSID.

Table 12-2. MSID Information

Width (Bytes)	Field Name	Type	Values	Description
5	Build Data Type	Char	If Processing Flag = UN, all data types in MSFC-STD-1274B, Volume 2 are supported. If Processing Flag = CO or CA, data types in Table 10-1 are supported.	Specifies the unprocessed, converted, or calibrated datatype that is associated with the MSID that is being sent in the GSE Packet. For more information on the Data Types, See Appendix B Data Types in MSFC-STD-1274B, Volume 2.
6	Build Sample Rate	Integer	1 or maximum sample rate	Indicates the maximum sample rate of the MSID or indicates value of 1 if only First Sample was selected for Sample Flag.
4	Build Length	Integer	If the first character of the build data type is “Sxxxx” for string, then the length is in bytes, otherwise the length is in bits.	Indicates the length of the MSID in bits or bytes, depending upon the build data type of the MSID.
10	Build Engineering Unit	Char	See Appendix D, of MSFC-DOC-1949, Volume 4 for valid Engineering Units	Indicates the Engineering Unit of the MSID.

The following is the description of how to use the GSE Packet Definition file to support processing of GSE packets, which are defined in Section 8.6. The GSE Packet Information section of the GSE Packet Definition file defined in Table 12-1 contains general information pertaining to the selected GSE Packet format. The GSE Packet ID is a unique identifier for the selected GSE packet. The format ID is used to differentiate this packet from other formats of the same GSE Packet ID. The Total Packet Length field provides the overall packet length, including all EHS and CCSDS headers. This field might be used to allocate memory for internal storage of the GSE packet. The Output Rate field indicates the rate, in packets per second, which the GSE packet will be transmitted by the POIC. The Description field provides an ASCII description of the GSE packet. The three Valid Database Version fields indicate the last three Telemetry Database versions that this GSE Packet had been validated against.

The MSID Information section of the GSE Packet Definition file is repeated for each MSID of the GSE Packet. Reference Figures 8-12 and 8-13 for graphical illustrations of the MSID structure in the GSE Packet. The MSID field provides the MSID ASCII name. The Start Octet field defines the octet of the GSE packet, relative to the beginning of the GSE packet, where the GSE Packet Overall Status field is located. The Overall Status field is the first field of each MSID information structure in the GSE packet. The Sample Flag field indicates whether one (first) or all samples of the MSID are included in the GSE packet. If Sample Flag is "F" (first), then the GSE Packet will contain one each of the following fields in addition to a single Overall Status field and a single Number of Samples field for the MSID:

Sample Status
Data Sample

If Sample Flag is "A" (all), then the GSE Packet will contain "M" number of the following fields, where M is the Build Sample Rate (defined below), in addition to a single Overall Status field and a single Number of Samples field for the MSID:

Sample Status
Data Sample

The Processing Flag field specifies whether all samples of the MSID are Calibrated, Converted, or Unprocessed. The Build Sample Rate field specifies the maximum sample rate for this MSID. A value of "1" will indicate that the "First Sample" was selected. The Build Datatype specifies the unprocessed, converted, or calibrated datatype of the MSID Data Sample(s). The Build Length field specifies the length of the MSID Data Sample in bits, or bytes. If the Build Datatype is one of the "String" datatypes (i.e. Build Datatype field begins with an "S"), then the Build Length is in bytes. Otherwise, the Build Length is in bits.

> Note: For unprocessed MSIDs only, the Build Length will specify the number of "meaningful" bits that will be transferred in the GSE packet, i.e. the raw measurement data received from the ISS. However, the GSE Packet Data Sample field size (see Figure 8-12) is rounded up to the nearest 8-bit byte. The unprocessed data will be right-justified in the GSE Packet Data Sample field. For example, if an unprocessed MSID's length is 21 bits, the Build Length field will be set to 21. The GSE Packet Data Sample field size will be 24 bits. The 21 bit measurement will be right-justified in the GSE Packet Data Sample field and the remaining 3 bits will be zero-filled.

Finally, the Build Engineering Unit field specifies the engineering unit of the MSID.

13.0 FILE TRANSFER

File transfer applications are used to send and retrieve files from a POIC-provided workstation or from the PIMS repository. Files can be transferred using the file transfer capability on a POIC-provided workstation using FTP software from a remote user GSE (TBR 13-1), or using the PIMS application software for files residing on or being sent to the PIMS repository. FTP procedures have been developed to assist the remote user in using the POIC-provided file transfer capability. See https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page for the detailed procedures. This section describes the scenarios, configurations, and limitations associated with the file transfers.

13.1 POIC PROVIDED WORKSTATION FILE TRANSFER SCENARIOS AND CONFIGURATIONS

There are four configurations that support file transfer operations between POIC-provided workstations and the remote user's GSE. Note that the NDE server is considered to be a POIC-provided workstation in the following configurations.

13.1.1 REMOTE USER'S GSE TO POIC PROVIDED WORKSTATION THROUGH POIC FILE TRANSFER INBOUND

Configuration #1: Get from a remote user's GSE to POIC-provided workstation using X-Windows, POIC File Transfer application, and FTP. The process is:

a. A remote user on GSE accesses a POIC-provided workstation through X-Windows. (See Section 5.)

b. From the Launchpad, the user initiates the File Transfer application under the Utilities pulldown menu.

c. The user specifies the username/password for the remote user's GSE and the directory path on the remote user's GSE for the file to be pulled into the POIC.

d. The user specifies the file type for the file to be stored from a list of restricted file types. (See Section 13.4 File Import Restriction.)

Remote user GSE/ user requirements:

a. The remote user's GSE must be approved through the HOSC account request form for X-Window access and FTP inbound through the Firewall.

b. The remote user's GSE must support FTP server functionality.

c. The remote user must be approved through the HOSC account request form for the File Transfer capability on the Launchpad.

d. The remote user must have an active account on the POIC-provided platform the user is accessing.

13.1.2 POIC PROVIDED WORKSTATION TO REMOTE USER GSE THROUGH POIC FILE TRANSFER OUTBOUND

Configuration #2: Push from POIC provided workstation to a remote user's GSE using X-Windows, POIC File Transfer application, and FTP. The process is:

a. A user on a remote-user GSE accesses a POIC-provided workstation through X-Windows (see section 5).

b. From the Launchpad, the user initiates the File Transfer application under the Utilities pulldown menu. (See POIC Capabilities Document (SSP 50304), section 4.7.7.)

c. The user specifies the file on the POIC-provided workstation to be transferred.

d. The user specifies the username/password for the remote user's GSE and the directory path for the file to be stored on the remote user's GSE.

Remote user's GSE/user requirements:

a. The remote user's GSE must be approved through the HOSC account request form for X-Windows access through the Firewall.

Note: Outbound FTP through the POIC Firewall is not restricted.

b. The remote user's GSE must support FTP server functionality.

c. The remote user's firewall must allow in-bound ftp connection requests.

d. The remote user must be approved through the HOSC account request form for the File Transfer capability on the Launchpad.

e. The remote user must have an active account on the POIC-provided platform the user is accessing.

13.1.3 REMOTE USER GSE FILE TRANSFER TO POIC PROVIDED WORKSTATION THROUGH FTP INBOUND (TBR13-1)

Configuration #3: Push from a remote user's GSE to the POIC-provided workstation using FTP and POIC File Import. The process is:

a. A user on a remote user's GSE uses the local FTP utility to initiate an FTP session with a POIC-provided workstation.

b. The user specifies the file on the local remote user's GSE to be transferred.

c. The user specifies the username/password for the POIC-provided workstation and the directory path for the file to be stored on the POIC-provided workstation. The user will be restricted to a holding area on the workstation, and will not be allowed to place the file in any directory that POIC software uses.

d. The user must set permissions on the file such that a POIC user can access the file through the POIC File Import capability on the Launchpad. The File Import capability is restricted to storing files on the POIC-provided workstation that are compliant with section 13.2.

e. FTP Commands allowed to the remote user while in the POIC holding area include:

   (1) Put

   (2) Change owner

   (3) List directory

   (4) Delete files

Remote user's GSE/ user requirements:

a. The remote user's GSE must be approved through the HOSC account request form for FTP inbound access through the POIC Firewall.

b. The remote user must have an active account on the POIC-provided platform the user is accessing.

c. Remote user GSE must have FTP client software.

13.1.4 POIC PROVIDED WORKSTATION TO REMOTE USER GSE FILE TRANSFER THROUGH FTP OUTBOUND (TBR 13-1)

Configuration #4: Pull from the POIC-provided workstation to a remote user's GSE using FTP and POIC File Export capability. The process is:

a. An internal POIC user on a POIC-provided workstation must have placed the file in the holding area from the POIC File Export capability and set permissions such that a remote user can access the file prior to when a remote user attempts access.

b. A user on a remote user's GSE uses the local FTP utility to initiate an FTP session with a POIC and POIC-provided workstation.

c. The remote user specifies the username/password for the POIC-provided workstation and the directory path for the file to be retrieved from the POIC-provided workstation. The remote user will be restricted to a holding area on the workstation from which to retrieve files.

d. The remote user specifies the directory path on the remote user's GSE where the file is to be stored.

e. FTP Commands allowed by the remote user while in the POIC holding area are:

   (1) Get

   (2) List directory

   (3) Delete files

Remote user's GSE/user requirements:

a. The remote user must have an active account on the POIC-provided platform the user is accessing.

b. Remote user GSE must support FTP client software.

13.2 POIC PROVIDED WORKSTATION FILE TRANSFER LIMITATIONS

The filename length of files used in the POIC software is restricted to 24 characters total. The 24-character limitation includes a 4-character extension. The user supplies the first 20 characters when naming a file and the system provides the extension. The extension includes the period (.) and three characters. If the extension is less than 3 characters, the filename is still restricted to the user supplied 20 characters. When a filename is read into the POIC, the filename will be used up until the first period. If a file is transferred into the POIC that is longer than the 20-character user-supplied filename, the file transfer application will truncate the filename after the 20th character and add the 4-character extension. The exception for this is word processing and spreadsheet files since these files are used by COTS applications, not POIC-developed software.

Section 13.4, File Import Restrictions, lists the extensions for the files that are imported into the POIC. A remote user may enlist the aid of an Integrated Support Team member to retrieve files from remote user GSE of a type normally restricted.

Files are transferred using FTP as defined in section 3.3.2.4.1. The following sections give further limitations on the file formats of some of the file interfaces defined at the POIC.

13.2.1 COMPUTATION SOURCE FILE FORMATS

A computation source file can be generated at a remote site and transferred to the POIC to be compiled and run. Users can retrieve the computation source files to the POIC provided workstation to be compiled on that local machine. The format of computation source files generated at a remote site must adhere to HOSC-EHS-125 to successfully execute in the POIC. For the required format and content of the Computation source files, see HOSC-EHS-125, Using the HOSC Computation Generation/Operation Software.

13.2.2 COMPUTATION INPUT/OUTPUT FILE FORMATS

The format of the computation output files is identical to the Standard output file described in section 9.1. Both the Standard output files and the computation output files can also be read into a POIC computation. Thus, the format of the computation input files is also identical to the Standard output file.

A Standard output file (or input file) for a computation can be generated at a remote site and transferred to the POIC for computation execution. The computation will process up to the "#End_Data" marker in the file and then stop reading the file. Any information which is available after the "#End_Data" marker is for viewing purposes only. For details on what is available in the information section of the Computation Output File Format, see HOSC-EHS-125, Using the HOSC Computation Generation/Operation Software user guide.

13.2.3 SCRIPTING FILES

A script is an interpreted language that receives telemetry input data, either unprocessed or pseudo data, and performs operations on the input data from the POIC. A script can be generated at a remote site and transferred to the POIC to be run after it has been validated. Users can retrieve the script source files to the POIC-provided workstation to be validated on that local machine. The format of script source files generated at a remote site must adhere to HOSC-EHS-1126 to successfully execute in the POIC. For the required format and content of the input and output files for the scripting services, see the Scripting user's guide HOSC-EHS-1126, Using the HOSC Script Generation/Operation Software.

13.3 PIMS FILE TRANSFER

The Payload Information Management System (PIMS) allows a remote user to transfer files into and out of the POIC using the PIMS Server (PIMS Repository). PIMS also allows the remote user to move files from the PIMS Server to another remote computer. The PIMS architecture for transferring files is depicted in Figure 13-1.

The Common Object Request Broker Architecture (CORBA) is used to transfer files between a remote PIMS client and the PIMS Server. CORBA allows the PIMS client to use a messenger, known as an Object Request Broker (ORB), to request file transfer services from the PIMS Server. ORB requests and return replies are exchanged across the PIMS web interface via the Internet Inter-ORB Protocol (IIOP). For more information on CORBA and IIOP, see sections 3.3.2.4.7 and 3.3.2.4.7.1.

The FTP protocol is used whenever a PIMS client requests that files be transferred from the PIMS Repository to/from a 3rd party remote computer. The 3rd party computer does not need to have a PIMS account, but must be able to act as an FTP server and must have a POIC-approved IP address for transferring files via FTP. FTP sessions with a 3rd party computer must be initiated from the PIMS Server. A computer outside the POIC firewall cannot initiate an FTP session with the PIMS Server.

If the 3rd party computer shown in Figure 13-1 is a PIMS client, that client would also use CORBA to send/receive files to/from the PIMS Server. To move a file from one client to another, the first client would place the file on the PIMS Server via CORBA while the second client waits. The second client would then request the file via CORBA.

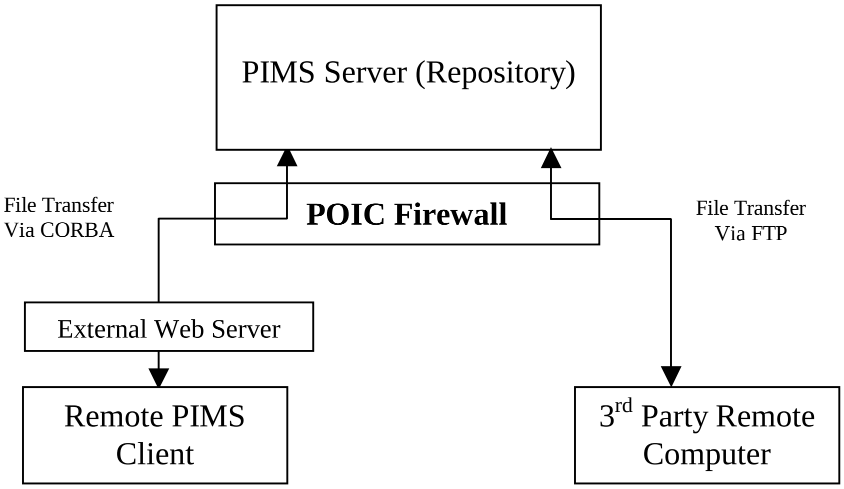


Figure 13-1. PIMS File Transfer Protocols

A remote user must do the following to access PIMS services:

a. Establish an EHS account with PIMS privileges on the POIC External Web server.

b. Install the required Web browser and associated Java Archive (JAR) files (see section 6.0). In addition, the remote user must install the POIC-provided digital certificate (see section 4.5).

For more information on the PIMS applications, see HOSC-EHS-136, Using the HOSC PIMS Software.

13.3.1 PIMS DOCUMENT FILE FORMATS

A user can create a document on the desktop and store it to PIMS using the store capability provided to the remote user within PIMS. For document modification, the user can retrieve the document to the desktop, update the document, and store it back to PIMS when completed. Any PIMS user can view the document; however, the document can only be edited by the original author.

Documents that are stored in PIMS go through a virus scan to ensure no viruses are found in the document. If these documents are encrypted when stored in PIMS, virus scanning will miss a virus in the document. The remote user can store the following document file formats in PIMS:

a. ASCII

b. Deleted

c. Deleted

d. Microsoft Word ™

e. Microsoft Excel ™

f. Portable Document Format (PDF)

13.3.2 UPLINK FILE FORMATS

Files to be uplinked to the ISS via the POIC must meet ISS vehicle limitations on file size and file name. File naming restrictions, including file name length, are enforced through the PIMS uplink file registration process. PIMS registration requires that the file name be unique within a Mission, Operational Support Mode, and Project (MOP), and 24 characters or less in length. The registered file name will be the onboard file name for files intended for transfer to the ISS. Revision information is not part of the file name registered in PIMS or used within the POIC.

The maximum length of a file itself is restricted to 8 Megabytes, and the file must end on an even byte boundary. File format is unrestricted. Uplink files can be ASCII files, binary files, or any other file type required by the user's on-board payload. If the uplink file is destined for a payload support system, a specific file format may be required. These file formats will be documented in the POIF Joint Operations Procedures or other equivalent agreements.

A remote PIMS client transfers uplink files to the PIMS Repository via CORBA. If the client X-windows into an EHS workstation, the file can be transferred to that workstation via FTP.

13.3.3 TIMELINER SOURCE FILE FORMAT

Each Timeliner procedure is created in an ASCII form referred to as the Bundle Source File. The file and a list of compilation options is input into the Timeliner Compiler that verifies syntax, resolves external references, and generates output files. Each Bundle Source File is created according to the User Interface Language Specification (SSP-30539). A remote PIMS client transfers the Timeliner Source File to the PIMS Repository via CORBA. If the client X-windows into an EHS workstation, the file can be transferred to that workstation via FTP.

13.3.4 DELETED

13.4 POIC FILE IMPORT RESTRICTIONS

In synopsis, Table 13-1 gives the specific information about each type of file that can be imported to the POIC. A description of each column in Table 13-1 is provided below.

File Type/Name: A basic name of the file type imported to the POIC.

File Description: A basic description of the type of file.

File Name Size: The maximum number of characters allowed in the file name. The standard being the 20 characters with a 4-character extension (for the period and normally 3 letters) giving a 24-character maximum.

Filename Extensions: Specifies the POIC provided extension or the vendor provided extension on files.

File Size Limits: The maximum size, if applicable, that the file supports.

File Format Definition:

1. **Vendor Proprietary:** Name of product will be given.
2. **POIC Documented File Format:** A structured file format with reference to the specific POIC to Generic User IDD Section or User's Guide for format definition.
3. **Unrestricted:** Any file type is acceptable, such as ASCII files or binary files.
4. **POIC Undocumented File format:** This format style may include a reference to the capabilities or information within the file, but is not guaranteed to establish format because it is only understood by POIC software.
5. **Industry Standard:** Accepted file extension(s) is given.

Transfer Protocol: FTP or CORBA

Table 13-1. Importable POIC Files

File Type/Name	File Description	File Name Size	File Name Extension	File Size Limits	File Format Definition	Transfer Protocol	
						To/From EHS WS*	**To/From PIMS Server**
Computation MSID Input List	Correlates MSIDs with variables as input to comp	20 char + 3 char extension	.in	None	POIC Documented File Format HOSC-EHS-125	FTP	N/A
Computation MSID Output List	Correlates MSIDs with variables as output from comp	20 char + 4 char extension	.out	None	POIC Documented File Format HOSC-EHS-125	FTP	N/A
Computation Source	Source code format for Computation in C or FORTRAN	20 char + 4 char extension	.c (C source code) .for (FORTRAN source code)	None	POIC Documented File Format HOSC-EHS-125	FTP	N/A
Display Files	POIC generated format to view telemetry data graphically	20 char + 4 char extension	.dsp	None	POIC Undocumented File format HOSC-EHS-124	FTP	N/A
Pixmap Files	Standard bitmaps or pixmap files to use in Displays	Only 24 characters visible at POIC	.xpm (pixmap files) .xbm (bitmap files)	None	Industry Standard format .Xpm, .Xbm	FTP	N/A
Script Source Files	POIC generated format for interpretive code	20 char + 4 char extension	.src	None	POIC Documented File Format. HOSC-EHS-1126	FTP	N/A

Table 13-1. Importable POIC Files

File Type/Name	File Description	File Name Size	File Name Extension	File Size Limits	File Format Definition	Transfer Protocol	
						To/From EHS WS*	**To/From PIMS Server**
Standard Output Files	Output format of data from comps or stored parameter reports	20 char + 4 char extension	.sto	None	POIC Documented File Format SSP 50305, Volume 1,section 9.1	FTP	N/A
Timeliner Source	ISS format for on-board timeliner source code	Only 24 characters visible at POIC		8 Megabytes	ISS Unique Format:	FTP	FTP or CORBA
Uplink Files	Files sent from the ground up to the ISS	24 characters, including file extension		8 Megabytes . File must end on an even byte boundary.	Unrestricted	FTP	FTP or CORBA
GSE Command Load Files	Allows GSE user to update modifiable commands in the Operational Command Database	20 char + 4 char extension	.gse	None	POIC Documented File Format. SSP 50305, Volume 2, section 4.0	FTP	N/A
Microsoft Word	Word processing files.	Only 24 characters visible at	.doc	8 Megabytes if uplinked	Vendor Proprietary - Microsoft	N/A	FTP or CORBA

Table 13-1. Importable POIC Files

File Type/Name	File Description	File Name Size	File Name Extension	File Size Limits	File Format Definition	Transfer Protocol	
						To/From EHS WS*	**To/From PIMS Server**
		POIC		to ISS			
Microsoft Excel	Spreadsheet files.	Only 24 characters visible at POIC	.xls	8 Megabytes if uplinked to ISS	Vendor Proprietary - Microsoft	N/A	FTP or CORBA
Portable Document Format	File format for viewing and navigating electronic images of often graphics-intensive hardcopy documents.	Only 24 characters visible at POIC	.pdf	8 Megabytes if uplinked to ISS	Vendor Proprietary - Adobe	N/A	FTP or CORBA
ASCII Files	A standard text file format. Possible characters for an ASCII file are shown in Appendix E.	Only 24 characters visible at POIC	.txt is most common	8 Megabytes if uplinked to ISS	Industry Standard Format. Typically uses a .txt file extension.	FTP	FTP or CORBA

*See 50304, section 4.7.7 for more information on file transfer to an EHS workstation via FTP. Note that section 4.7.7 refers to an EHS workstation as a “Real Time Data System (RTDS) Workstation”.

14.0 E-MAIL SERVICES

The electronic mail (e-mail) service uses the SMTP (RFC 821) method to transfer e-mail messages. The domain name for accessing the POIC is “mps.hosc.msfc.nasa.gov.” The POIC will not forward any mail received to another mail server. The format for sending e-mail to users in the POIC from outside the Mission PC Services (MPS) Firewall is “<UserID>@mps.hosc.msfc.nasa.gov”.

There is a maximum contiguous file size of five megabytes for an attachment that will be accepted by the POIC. The restrictions on the type of files that can be attached to the mail message are based upon the privilege that a user has to access a file.

An internal POIC user may receive a message with attachments. The attachment may be opened directly from the Mail application if the file type is recognized and allowed. Messages and attachments received by an internal POIC user are located on the Mail Server.

This Page Intentionally Left Blank

15.0 PAYLOAD PLANNING SYSTEM SERVICES

The details of how the Payload Planning System Services will exchange information between the International Partner Control Centers and the POIC will be documented in the Multilateral Distributed Planning Interface Specification (SSP 50401).

This Page Intentionally Left Blank

16.0 TELESCIENCE RESOURCE KIT (TREK) INTERFACES

The TReK is one of the HOSC EHS Remote Operations configuration options. TReK consists of a PC configured with COTS software, shareware, freeware, MSFC Mission Systems Development Group developed software, and POIC-provided interface software to provide the same basic functions as a POIC-provided workstation. For more information about the TReK capabilities offered to the remote user, see POIC Capabilities Document (SSP 50304), section 6.

The intent of this section is to provide information regarding the TReK API. Users can extend TReK capabilities by using the TReK API together with COTS products to use local telemetry and command functions. The TReK API provides a way to build connections between a favorite COTS product and local telemetry and command functions, such as retrieving real-time data and uplinking commands. This provides a way to enhance COTS products by adding telemetry and command functionality. For example, the user might want to use G2 to build an expert system that executes a command based on a specific situation.

The TReK API is documented in the following document:

Name:	Telescience Resource Kit (TReK) Application Programming Interface Reference Manual
Document Number:	MSFC-DOC-2800

This Page Intentionally Left Blank

17.0 MANUAL PROCEDURES VIEWER

Manual Procedure Viewer (MPV) software allows users to view the Payload Operation Data File (PODF). The PODF is a file of manual procedures to be followed by the ISS crew in managing payload experiments on ISS. It is a hypertext-linked collection of test procedures, images, and animations for use by the crew. Remote users can access the MPV software by using a Virtual Private Network connection to the MPV servers in the POIC. Remote users gaining access to MPV will be required to use Checkpoint's SecuRemote software, which is IPSec-compliant. SecuRemote software will be bundled up with other necessary software to access MPV resources. The installation procedures for establishing a VPN with the POIC can be found at https://aristotle.hosc.msfc.nasa.gov/PGUIDD_Page. Once the VPN software has been installed, procedures for accessing the MPV software can be found at the same URL**.** More information regarding IPSec and required ports to support IPSec can be found in section 3.3.2.4.8 .

This Page Intentionally Left Blank

APPENDIX A: ACRONYMS

AMI	Alternate Mark Inversion
AOS	Acquisition of Signal
API	Application Programming Interface
APID	Application Process Identifier
BPDU	Bitstream Protocol Data Unit
B8ZS	Binary 8 Zero Substitution
C&C	Command and Control
CCSDS	Consultative Committee for Space Data Systems
CDD	Context Dependent Decommutation
CDP	Custom Data Packet
CDR	Common Data Representation
CIO	Chief Information Officer
COR	Communications Outage Recorder
CORBA	Common Object Request Broker Architecture
COTS	Commercial-Off-the-Shelf
CPU	Central Processing Unit
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CSO	Computer Security Official
CSS	Command Systems Services
DES	Data Encryption Standard
DIX	Digital/Intel/Xerox
DNS	Domain Name Service
DP	Distribute Packets
DPA	Dual Phone Adapter

DQ	Data Quality
DS0	Digital Standard 0
DSS	Digital Signature Standard
DSID	Data Stream Identifier
DTA	Dual Trunk Adapter
EHS	Enhanced HOSC System
EIA	Electronic Industries Association
EML	Extract MSID Library
E&OI	Engineering and Operations Integration
ERIS	EHS Remote Interface System
ESF	Extended Super Frame
FDDI	Fiber Distributed Data Interface
FTP	File Transfer Protocol
GIOP	General Inter-ORB Protocol
GMT	Greenwich Mean Time
GRT	Ground Receipt Time
GSCB	Ground Segment Control Board
GSE	Ground Support Equipment
GSRT	Ground Support Requirements Team
HMCG	HOSC Management Coordination Group
HOSC	Huntsville Operations Support Center
HRDS	High Rate Data System
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HUA	HOSC User Assistance

ICD	Interface Control Document
ICMP	Internet Control Message Protocol
IDD	Interface Definition Document
IDL	Interface Definition Language
IEEE	Institute of Electrical and Electronics Engineers
IGSS	International Ground System Specification
IIOP	Internet Inter-ORB Protocol
IP	Internet Protocol
IRIG-B	Interrange Instrumentation Group, Standard B
ISS	International Space Station
JAR	Java Archive
JDK	Java Development Kit
JRE	Java Runtime Environment
JSC	Johnson Space Center
JVM	Java Virtual Machine
LAN	Local Area Network
LES	Limit/Expected State
LLC	Logical Link Control
LOR	Line Outage Recorder
LOS	Loss of Signal
LSB	Least Significant Bit
MAC	Media Access Control
MBF	Mission Build Facility
MDM	Multiplexer/Demultiplexer

MET	Mission Elapsed Time
MLP	Multiline Phone
MOL	Mission Operations Laboratory
MOP	Mission, Operational Support Mode and Project
MPEG	Motion Pictures Experts Group
MPS	Mission PC Services
MSB	Most Significant Bit
MSFC	Marshall Space Flight Center
MSID	Measurement Stimulation and Identifier
MSN	Mission Systems
NASA	National Aeronautics and Space Administration
NASCOM	NASA Communications
NDE	Nonoperational Development Environment
NISN	NASA Integrated Services Network
NRT	Near Real Time
NTSC	National Television Standards Committee
OD	Operational Data
OMG	Object Management Group
ORB	Object Request Broker
PCDB	Project Command Database
PDF	Portable Document Format
PDL	Payload Data Library
PDSS	Payload Data Services System
PHY	Physical Layer Protocol
PIMS	Payload Information Management System

PMD	Physical Medium Dependent
POIC	Payload Operations Integration Center
POIF	Payload Operations Integration Facility
PPS	Payload Planning System
PTDB	Project Telemetry Database
PTT	Push to Talk
PUB	Publication
RFC	Request for Comments
RGB	Red, Green, and Blue
RPSM	Retrieval Processing Summary Message
RTDS	Real Time Data System
SAS	Single Attach Station
SF	Standard Frame
SMT	Station Management
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SSH	Secure Shell
SSL	Secure Socket Layer
SSL3	Secure Socket Layer 3
TBD	To Be Determined
TBR	To Be Resolved
TCP	Transmission Control Protocol
TDB	Telemetry Database
TDRSS	Tracking and Data Relay Satellite System
TDS	Time Distribution System

TLM	Telemetry
TNS	Telemetry Network Services
TReK	Telescience Resource Kit
TSC	Telescience Support Center
UDSM	User Data Summary Message
UDP	User Datagram Protocol
UOA	User Operations Area
URL	Universal Resource Locator
U.S.	United States
USOC	United States Operations
USOS	United States On-Orbit Segment
VCDU	Virtual Channel Data Unit
WS	Workstation
WSC	White Sands Complex

APPENDIX B: GLOSSARY

Applet
A piece of Java code that is transported from the Web Server to the remote user's GSE upon initiating the request for that capability on the POIC home page.

Application Process Identifier
An 11-bit field in the CCSDS primary packet header that identifies the source-destination pair for ISS packets. The type bit in the primary header tells you whether the APID is a payload or system source-destination pair.

Application Programming Interface
A set of functions used by an application program to provide access to a system's capabilities.

Authentication
The act of validating the claimed identity of a "subject" (e.g., individual, station, or originator). Normally, authentication follows the process of identification. This term is also used to describe the act of verifying the integrity of information that has been stored, transmitted, or otherwise exposed to possible unauthorized modification.

Bitstream Protocol Data Unit (BPDU)
A CCSDS protocol data unit of the bitstream function having a format of a header followed by a fixed length block of contiguous bitstream data.

Calibration
The transformation of a parameter to a desired physical unit (e.g., volts) or text state code.

Communications Outage Recorder
System that captures and stores payload science, health and status, and ancillary data during TDRSS zone of exclusion.

Computer Security Officer
Individual (designated in writing) assigned security and/or network security responsibility for a particular system or Data Processing Installation (facility).

Consultative Committee for Space Data Systems (CCSDS) Packet
A source packet comprised of a 6-octet CCSDS defined primary header followed by an optional secondary header and source data that together may not exceed 65,535 octets.

Conversion
Transformation of downlinked spacecraft data types to ground system platform data types.

Custom Data Packet
A packet containing a subset of parameters that can be selected by the user at the time of request.

Decommutation
Extraction of a measurement or parameter from telemetry.

Digital certificate
Public key certificates and digital signatures that allow parties who were previously unknown to each other to establish trust relationships and/or conduct secure, encrypted communications.

Dump
Onboard data that have been recorded and are downlinked at not necessarily the native rate.

Enhanced HOSC System (EHS)
Upgraded support capabilities of the HOSC systems to provide multifunctional support for multiple projects. EHS incorporates all systems required to perform data acquisition and distribution, telemetry processing, command services, database services, mission support services, and system monitor and control services.

Expected State Sensing
Process of detecting a text state code parameter in an off-nominal state.

Flight Ancillary Data
A set of selected core system data and payload health and status data collected by the USOS Payload Multiplexer/Demultiplexer (MDM). The data is used by experimenters to interpret payload experiment results.

Front End Workstation
The server that provides the user access to the NDE environment.

Ground Ancillary Data
A set of selected core system data and payload health and status data collected by the POIC that are used by experimenters to interpret payload experiment results. Ground Ancillary Data can also contain computed parameters (pseudos).

Ground Receipt Time
Time of packet origination. The time when the IRIG-B time signal is received.

Ground Support Equipment (GSE)
Equipment owned and maintained by the user rather than the POIC.

Ground Support Equipment Packet
A CCSDS packet that contains data extracted from any of the data processed by the POIC. The format of the packet is defined in the POIC telemetry database.

Health and Status Data
A set of data indicating the state and conditions of a system or payload.

Huntsville Operations Support Center (HOSC)
A facility located at the Marshall Space Flight Center (MSFC) that provides users the tools necessary for monitoring, commanding, and controlling various elements of space vehicle, payload, and science experiments. Support consists of real-time operations planning and analysis, inter and intracenter ground operations coordination, facility and data system resource planning and scheduling, data systems monitor and control operations, and data flow coordination.

Integrity
The state that exists when computerized data are the same as those in the source documents or have been correctly computed from source data and have not been exposed to accidental or malicious alteration or destruction.

Interface Control Document
Addresses all operational interface requirements between two or more users/facilities.

Interface Definition Document
Describes the standard interfaces between the two or more users/facilities.

Internet Protocol
Protocols supported by the Internet Engineering Task Force (IETF) that support connection-less computer communication methods.

Interrange Instrumentation Group -B (IRIG-B)
IRIG Timing standard with 1-second timing defined in IRIG Standard 200-95, IRIG Serial Time Code Formats. The POIC uses IRIG-B122 format.

Java Archive (JAR) File
When a Web page is sent to a user's computer, the page often includes Java applets. Applets are small applications that allow the user to interact with the Web page (perform calculations, execute animations, etc.). Applets include a set of required files that are transmitted to the user along with the applet executable file, e.g. image, sound, and *.class* files. These files are transmitted from the source Web server to the user via separate HTTP transactions, one for each file. The Java Archive (JAR) file format provides a way to bundle the applet and associated files into a single file that can be downloaded to the user via a single HTTP transaction, improving the speed of Web page downloads. The JAR format also supports compression, which reduces file size and further improves download speed.

Launchpad
An application within the EHS that is used to initiate all the POIC applications that the user is authorized to access.

Least Significant Bit (LSB)
The numeric order for the labeling of all bits within bytes/words is from Most Significant Bit to Least Significant Bit, meaning the LSB is bit 0. The low order bits of a data stream or channel.

Limit Sensing
Process of detecting caution and warning conditions for a parameter with a numerical value.

Line outage recorder playback
A capability provided by the White Sands Complex (WSC) to play back tapes generated at WSC during ground system communication outages.

Local table
A user copy of the real-time telemetry database which contains information regarding conversion, calibration, and limit sensing.

Malicious Code
Unauthorized subverting programs or subverting code that has been introduced into authorized software with the intent to, and purpose of, causing damage to data, applications, or networks. Malicious code includes viruses, time bombs, logic bombs, Trojan horses, and worms.

Measurement Stimulus Identifier (MSID)
An identifier that corresponds to the location of data within telemetry packets.

Mission, Operational Support Mode Project (MOP)
A MOP refers to a unique activity that can be supported by an RTDS. The term "mission" refers to a unique time period in a project’s lifecycle. For ISS, "mission" is associated with a unique increment. The term "operational support mode" refers to the type of activity occurring in a project’s lifecycle, including real-time mission support, verification and validation, ground system test and checkout of flight systems, simulation, training, and development. The term "project" describes a specific NASA project, such as the ISS, the Space Shuttle, and the Advanced X-ray Astrophysics Facility (AXAF).

Mission Systems (MSN)
Mission systems are those that control or directly support human space flight or unmanned space flight. If information in this category is lost, altered, or unavailable, the impact on NASA would be significant. The impact would result in the loss of major or unique tangible assets and may pose a threat to human life. The major IT security concerns that must be considered are integrity and availability. Confidentiality is usually of less concern; however, if confidentiality becomes a concern, then additional controls must be imposed to accommodate that concern.

Most Significant Bit (MSB)
The high order bit of a data stream or channel.

Multiplexed Protocol Data Unit
An intermediate unit between VCDUs and CCSDS packets. There may be multiple data packets in an MPDU.

NASA Integrated Services Network (NISN)
A NASA organization that provides Program Support Communications Network (PSCN), NASA Science Internet (NSI), and NASA Communications (NASCOM) services.

Nonoperational Development Environment (NDE)
An off-line (no mission impact) capability that users to generate necessary UDEs and to populate the databases in preparation for future flight activities.

On-orbit Data
Real-time data that are received directly from the spacecraft.

Operational Support Mode
The mode or type of operation that the POIC is currently configured to support. Possibilities include real-time (flight) and offline support. See Primary EHS Header Protocol for full listing of operational support modes used in EHS.

Packet Sequence Error
Indicates if a Packet Sequence Error was detected. Used to indicate that packet data were lost.

Payload Data Library (PDL)
An application that provides the interface for the user to specify which capabilities and requirements are needed to command and control their payload.

Payload Data Services Systems (PDSS)
A data system that receives, processes, stores and distributes ISS telemetry to the EHS, USOC, IPs, TSCs and other payload unique facilities.

Payload Information Management Systems (PIMS)
An EHS subsystem that provides an electronic information management system used by payload users, POIC Cadre, and others for increment preparation and planning.

Payload Operations Integration Center (POIC)
Manages the execution of on-orbit ISS payloads and payload support systems in coordination/unison with distributed International Partner Payload Control Centers, Telescience Support Centers (TSCs), and payload-unique remote facilities. The POIC resides within the HOSC.

Payload Planning Systems (PPS)
A planning tool that provides the interface for remote users and facilities to schedule activities and required resources associated with the payload.

PDSS Retrieval Processing
Provides the capability to retrieve payload data from storage. Retrieval processing may also include re-sequencing the payload data and removing redundancies and generating data quality information and placing it in the EHS protocol. The activity will be performed in non-real-time and will nominally always be available.

Playback
Data retrieved from some recording medium and transmitted to one or more users.

Private Data
Information that is specified as private in the POIC telemetry database so that only that owner is authorized to access the data.

Programmatic Interface
A software interface between computer systems. Unlike the API, functions will not be provided to interface with the POIC software. The user will be required to develop software to interact with programmatic services.

Pseudo - Telemetry (Pseudo Data)
Values that are created from calculations instead of directly transported telemetry data. The pseudo data can be created from computations or scripts and can be displayed on the local workstation. The data can be distributed if the pseudos are resident in the POIC Telemetry Database as an External pseudo.

Real Time
A data mode that processes data as they are received by the data system.

Remotely Generated Command
A command where the remote user generates the raw command bit pattern and sends it to the POIC for uplink to the ISS. This is in contrast to a "remotely initiated command" where the remote user simply sends a request to the POIC to uplink a command bit pattern that resides in the POIC Command Database.

Retrieval Processing Summary Message (RPSM)
A message that includes the size of the retrieval dataset, the time stamp indicating the time the dataset was processed, the dataset start and stop time, and the number of data packets included in the retrieval dataset.

Science Data
Sensor or computational data generated by payloads for the purpose of conducting scientific experiments.

Simple Mail Transfer Protocol (SMTP)
A platform independent transport protocol that utilizes TCP/IP to transfer mail between heterogeneous platforms.

Simulation
Represents an operation mode whereby specific resources are supporting users in a simulation configuration.

Telescience Support Center (TSC)
The TSC is a NASA funded facility that provides the capability to plan and operate on-orbit facility class payloads and experiments, other payloads and experiments, and instruments.

To Be Defined
Information is going to be provided but the details are not know yet.

To Be Resolved
An issue needs to be worked against a known requirement or a requested capability that has not been accepted as a baseline requirement.

Transmission Control Protocol/Internet Protocol
Protocol that ensures packets of data are delivered to destinations in the sequence in which the data were transmitted.

Uplink format
The bit pattern of the command or file to be uplinked.

User
Any individual with an interest in a payload or experiment on any increment of the ISS manifest that requires the use of capabilities or interfaces provided by the POIC in order to receive telemetry, send commands, use mission support services, or access to mission communications.

User Datagram Protocol (UDP)
UDP is a connection-less oriented protocol that does not guarantee delivery of data. In the TCP/IP protocol suite, the UDP provides the primary mechanism that application programs use to send datagrams to other application programs.

User Data Summary Message (UDSM)
Represents data quality reported by a data stream identifier for a particular user.

User defined data packets
Data packets that are formatted based upon user requests instead of ordered by the Telemetry Database.

User profile
A set of capabilities or privileges assigned to a user based upon his/her account. Used internally in the POIC to confirm that a user has access to various POIC functions.

United States Operations Center
ISS host facility in the HOSC where payload users can come to control and monitor their payloads instead of being located remotely.

Validation
Performance of tests and evaluations in order to determine compliance with security, specifications, and requirements.

Virtual channel data unit
A CCSDS data set of specific structure and of fixed length, which includes CCSDS-specified headers and into which user data are packaged for transmission over the space-to-ground link.

X-Windows Software
A software applications tool commercially available that allows a remote computer to have the same graphical user interface as the local computer system.

Web
Term used to indicate access through HTTP protocol; also referred to as the World Wide Web (WWW).

APPENDIX C: TBR ITEMS

To Be Resolveds (TBRs) are those items where a requirement has not yet been fully defined or resolved.

Section Number	TBR Item/Issue	STATUS
7.4.2.4 - Login Limit Status	**TBR 7-1:** Does the ERIS Server provide the capability to guarantee each TSC or Remote Site a minimum number of logins?	Open
12.1 - Partial Database File Format	**TBR 12-1:** Insufficient information is provided to the user for extraction and processing of individual parameters from a partial database file.	Open
12.1 - Partial Database File Format	**TBR 12-2:** The ASCII file names for the GSE Packet Definition File and partial telemetry database files have been defined, but the ASCII file names for the partial command database files have not.	Open. Available for Rev C of this IDD.
13.1.3 - Remote User GSE File Transfer to POIC-Provided Workstation Through FTP Inbound	**TBR 13-1:** At this time, FTP initiated from an external platform has been disabled due to the security concern of EHS passwords in the clear". Until this is fixed, FTP must be initiated from the inside outbound.	Open. Maybe resolved with the implementation of VPN.

This Page Intentionally Left Blank

APPENDIX D: TBD ITEMS

To Be Determineds (TBDs) are items where the requirement(s) has been agreed upon, but the full definition of the design is still under discussion.

Section Numbers	TBD Issue/Item	STATUS
8.5 - PDSS Retrieval Processing Summary Message (RPSM)	**TBD 8-1:** RPSM format, content, and utilization concept is TBD pending completion of the PDSS Data Storage Manager design.	Open
9-3 - Stored BPDU File Format	**TBD 9-1:** The stored BPDU file format has not been determined.	Open

This Page Intentionally Left Blank

APPENDIX E: ASCII CHART

The following is the ASCII Chart of the keyboard characters to Hexadecimal values. The subsequent chart shows both the hexadecimal numbers and the decimal numbers and the names of the abbreviations.

ASCII Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

ASCII Chart Detailed

VALUE	HEX VALUE	DECIMAL VALUE	NAME	ABBREV
<NU>	/x00	0	NULL	(NUL)
<SH>	/x01	1	START OF HEADING	(SOH)
<SX>	/x02	2	START OF TEXT	(STX)
<EX>	/x03	3	END OF TEXT	(ETX)
<ET>	/x04	4	END OF TRANSMISSION	(EOT)
<EQ>	/x05	5	ENQUIRY	(ENQ)
<AK>	/x06	6	ACKNOWLEDGE	(ACK)
<BL>	/x07	7	BELL	(BEL)
<BS>	/x08	8	BACKSPACE	(BS)
<HT>	/x09	9	CHARACTER TABULATION	(HT)
<LF>	/x0A	10	LINE FEED	(LF)
<VT>	/x0B	11	LINE TABULATION	(VT)
<FF>	/x0C	12	FORM FEED	(FF)
<CR>	/x0D	13	CARRIAGE RETURN	(CR)
<SO>	/x0E	14	SHIFT OUT	(SO)
<SI>	/x0F	15	SHIFT IN	(SI)
<DL>	/x10	16	DATALINK ESCAPE	(DLE)
<D1>	/x11	17	DEVICE CONTROL ONE	(DC1)
<D2>	/x12	18	DEVICE CONTROL TWO	(DC2)
<D3>	/x13	19	DEVICE CONTROL THREE	(DC3)
<D4>	/x14	20	DEVICE CONTROL FOUR	(DC4)
<NK>	/x15	21	NEGATIVE ACKNOWLEDGE	(NAK)
<SY>	/x16	22	SYNCHRONOUS IDLE	(SYN)
<EB>	/x17	23	END of TRANSMISSION BLOCK	(ETB)
<CN>	/x18	24	CANCEL	(CAN)
<EM>	/x19	25	END OF MEDIUM	(EM)
<SB>	/x1A	26	SUBSTITUTE	(SUB)
<EC>	/x1B	27	ESCAPE	(ESC)
<FS>	/x1C	28	FILE SEPARATOR	(IS4) or (FS)
<GS>	/x1D	29	GROUP SEPARATOR	(IS3) or (GS)
<RS>	/x1E	30	RECORD SEPARATOR	(IS2) or (RS)
<US>	/x1F	31	UNIT SEPARATOR	(IS1) or (US)
<SP>	/x20	32	SPACE	(SP)
<!>	/x21	33	EXCLAMATION MARK	
<">	/x22	34	QUOTATION MARK	
<&>	/x26	36	AMPERSAND	
<'>	/x27	37	APOSTROPHE	
<(>	/x28	38	LEFT PARENTHESIS	
<)>	/x29	39	RIGHT PARENTHESIS	
<*>	/x2A	40	ASTERISK	
<+>	/x2B	41	PLUS SIGN	

VALUE	HEX VALUE	DECIMAL VALUE	NAME	ABBREV
<,>	/x2C	42	COMMA	
<->	/x2D	43	HYPHEN-MINUS	
<.>	/x2E	44	FULL STOP	
<//>	/x2F	45	SOLIDUS	
<0>	/x30	46	DIGIT ZERO	
<1>	/x31	47	DIGIT ONE	
<2>	/x32	48	DIGIT TWO	
<3>	/x33	49	DIGIT THREE	
<4>	/x34	50	DIGIT FOUR	
<5>	/x35	51	DIGITFIVE	
<6>	/x36	52	DIGIT SIX	
<7>	/x37	53	DIGIT SEVEN	
<8>	/x38	54	DIGIT EIGHT	
<9>	/x39	55	DIGIT NINE	
<:>	/x3A	56	COLON	
<;>	/x3B	57	SEMICOLON	
<<>	/x3C	58	LESS-THAN SIGN	
<=>	/x3D	59	EQUALS SIGN	
</>>	/x3E	60	GREATER-THAN SIGN	
<?>	/x3F	61	QUESTION MARK	
<At>	/x40	62	COMMERCIAL AT	
<A>	/x41	63	LATIN CAPITAL LETTER A	
<B>	/x42	64	LATIN CAPITAL LETTER B	
<C>	/x43	65	LATIN CAPITAL LETTER C	
<D>	/x44	66	LATIN CAPITAL LETTER D	
<E>	/x45	67	LATIN CAPITAL LETTER E	
<F>	/x46	68	LATIN CAPITAL LETTER F	
<G>	/x47	69	LATIN CAPITAL LETTER G	
<H>	/x48	70	LATIN CAPITAL LETTER H	
<I>	/x49	71	LATIN CAPITAL LETTER I	
<J>	/x4A	72	LATIN CAPITAL LETTER J	
<K>	/x4B	73	LATIN CAPITAL LETTER K	
<L>	/x4C	74	LATIN CAPITAL LETTER L	
<M>	/x4D	75	LATIN CAPITAL LETTER M	
<N>	/x4E	76	LATIN CAPITAL LETTER N	
<O>	/x4F	77	LATIN CAPITAL LETTER O	
<P>	/x50	78	LATIN CAPITAL LETTER P	
<Q>	/x51	79	LATIN CAPITAL LETTER Q	
<R>	/x52	80	LATIN CAPITAL LETTER R	
<S>	/x53	81	LATIN CAPITAL LETTER S	
<T>	/x54	82	LATIN CAPITAL LETTER T	
<U>	/x55	83	LATIN CAPITAL LETTER U	

VALUE	HEX VALUE	DECIMAL VALUE	NAME	ABBREV
<W>	/x57	85	LATIN CAPITAL LETTER W	
<X>	/x58	86	LATIN CAPITAL LETTER X	
<Y>	/x59	87	LATIN CAPITAL LETTER Y	
<Z>	/x5A	88	LATIN CAPITAL LETTER Z	
<<(>	/x5B	89	LEFT SQUARE BRACKET	
<////>	/x5C	90	REVERSE SOLIDUS	
<)/>>	/x5D	91	RIGHT SQUARE BRACKET	
<'/>>	/x5E	92	CIRCUMFLEX ACCENT	
<_>	/x5F	93	LOW LINE	
<'!>	/x60	94	GRAVE ACCENT	
<a>	/x61	95	LATIN SMALL LETTER A	
<b>	/x62	96	LATIN SMALL LETTER B	
<c>	/x63	97	LATIN SMALL LETTER C	
<d>	/x64	98	LATIN SMALL LETTER D	
<e>	/x65	99	LATIN SMALL LETTER E	
<f>	/x66	100	LATIN SMALL LETTER F	
<g>	/x67	101	LATIN SMALL LETTER G	
<h>	/x68	102	LATIN SMALL LETTER H	
<i>	/x69	103	LATIN SMALL LETTER I	
<j>	/x6A	104	LATIN SMALL LETTER J	
<k>	/x6B	105	LATIN SMALL LETTER K	
<l>	/x6C	106	LATIN SMALL LETTER L	
<m>	/x6D	107	LATIN SMALL LETTER M	
<n>	/x6E	108	LATIN SMALL LETTER N	
<o>	/x6F	109	LATIN SMALL LETTER O	
<p>	/x70	110	LATIN SMALL LETTER P	
<q>	/x71	111	LATIN SMALL LETTER Q	
<r>	/x72	112	LATIN SMALL LETTER R	
<s>	/x73	113	LATIN SMALL LETTER S	
<t>	/x74	114	LATIN SMALL LETTER T	
<u>	/x75	115	LATIN SMALL LETTER U	
<v>	/x76	116	LATIN SMALL LETTER V	
<w>	/x77	117	LATIN SMALL LETTER W	
<x>	/x78	118	LATIN SMALL LETTER X	
<y>	/x79	119	LATIN SMALL LETTER Y	
<z>	/x7A	120	LATIN SMALL LETTER Z	
<(!>	/x7B	121	LEFT CURLY BRACKET	
<!!>	/x7C	122	VERTICAL LINE	
<!)>	/x7D	123	RIGHT CURLY BRACKET	
<'?>	/x7E	124	TILDE	
<DT>	/x7F	125	DELETE	(DEL)

APPENDIX F: SECURITY PLAN CHECKLIST

Security Plan for Remote Sites

Site Name: ______________________________ Date: ______________

Primary Point of Contact: ______________________ / ______________________
Type or Print Name Signature

Phone: ______________________

Email: ______________________

Computer Security Official: ______________________ / ______________________
Type or Print Name Signature

Phone: ______________________

Email: ______________________

See Table 4-3 for guidance:

		In-Place	Planned	(MO/YR)	N/A
(1.)	Security Management	( )	( )	__/__	( )
(2.)	Workstation Access Control	( )	( )	__/__	( )
(3.)	Configuration Management	( )	( )	__/__	( )
(4.)	Physical Access Controls	( )	( )	__/__	( )
(5.)	Network Access Controls	( )	( )	__/__	( )
(6.)	Personnel Security	( )	( )	__/__	( )
(7)	Audit Trails	( )	( )	__/__	( )
(8)	DB Management System Protection	( )	( )	__/__	( )
(9)	Information and Application Protection	( )	( )	__/__	( )

Additional Comments:

Attach additional pages if necessary

This Page Intentionally Left Blank